# SINCLAIR QL WORLD

Volume 3 Issue 3 Price £2.50

## New User Guide

How QDOS is divided up.

## Diskcopy

A disk backup routine.

## SuperBasic in Action

"Browser" file handling.

## The World at your fingertips

A World Map adapted from PD.

ISSN 0951-9335

03

9 770951 933023

# DIGITAL PRECISION

## THE POWER BEHIND THE BUTTON

# 1984-1994

## THE BEST SOFTWARE FOR THE BEST COMPUTER

**A PRIDE IN ITS LOGIC**

**NO IDEALISTIC PRIG**

**SPIRITED LOGICIAN**

**DIGITAL PRECISION**

**A LOGIC IN ITS PRIDE**

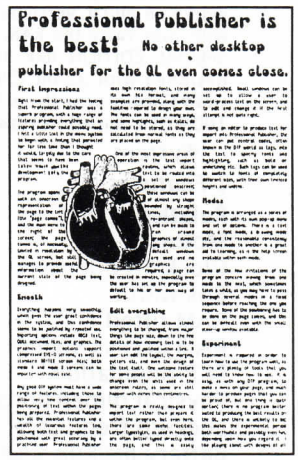
# PERFECTION SPECIAL EDITION

## POWER

PERFECTION SPECIAL EDITION has 253 (two hundred and fifty three) direct/menu commands (not counting options in sub-menus), plus 32 special characters (like Bold on) that can be inserted 'directly' plus intelligent (and now excellently documented) macros. Comparisons with other word processors on the subject of power are hence quite unnecessary.

## EASE OF USE

Independent reports, customer feedback and published reviews (of its less able but still excellent predecessor, PERFECTION) leave one in no doubt as to which word processor is friendliest – PERFECTION SPECIAL EDITION, with its intuitive, silky handling. Uniquely, it has two operating modes, with both menus (visible or invisible – they even look like Quill's) and direct commands (for when you familiarise yourself with the system). Uniquely, both modes are 're-entrant' (so you can use any menu option or direct command while you are in the middle of performing another option or command – block handling, etc, becomes a dream). Uniquely, PERFECTION SE has fully automatic memory management, grabbing and releasing RAM instantly as your document grows or shrinks – programs without this don't take full advantage of the multi-tasking abilities of the QL! Uniquely, PERFECTION SE leaves you in the driving seat, not juggling things around 'underfoot' while you are typing. Uniquely, PERFECTION SE allows up to nine different documents to be handled simultaneously from one copy of the program – each with totally independent margin, tab, justification, control panel, etc, settings. Uniquely, each document can itself have up to six environment settings, each settable or recallable instantly with a single keypress combination. Each document can have any number (up to 500,000 on GOLD CARD) of candidate blocks! Each document can have two independent windows (of any depth, of any (but same) width across) 'on to' it, even with overlapping text – that allows you to edit in one place while viewing another, to compare 'before editing' with 'after editing' (you can arrange to have one window remain 'frozen' in time), etc. Uniquely, we realise how much faster it is to type in something like CTRL/SHIFT/F5 than (say) F3 F3 R – both involve three keys, but as the former doesn't require the keys to be pressed in just one specific order, or to be released in any order at all (together will do), it is in practice twice as fast as the latter, where no key may be pressed until its predecessor is released. PERFECTION SE takes advantage of all this – it is the little things that count! Uniquely, by providing eight user-definable strips, PERFECTION SE allows you to cope with printers of the future, not just the printers that now exist – you can attach the strips to any printer features. Uniquely, PERFECTION SE's status lines give full information on all relevant global settings. And the manual has an index. Also, it has all the important bits at the front.


Professional Publisher is the best! No other desktop publisher for the QL even comes close.

First impressions

Smooth

Edit everything

Modes

Experiment

## WYSIWYG?

By the latest definition of this term, neither is PERFECTION SE fully WYSIWYG, nor are other QL word processors. WYSIWYG means what you see on screen is exactly what you get on paper. Exactly – down to every wiggle in every character in every font.

To get true WYSIWYG, use PERFECTION SE's fully automatic link (supplied as part of PERFECTION SE) to PROFESSIONAL PUBLISHER, where you will get 100% WYSIWYG. 100%? Yes, 100%. With this combination, adjust the horizontal and vertical magnification on your monitor (ie calibrate it once and for all so screen circles correspond to same-diameter printed circles – poor monitors may distort a little bit at the edges). Now you can place your printed output from PERFECTION via PUBLISHER over your monitor screen, and get a match that is more perfect than is your eyesight. Now that is WYSIWYG.

## SUPERB PRINT QUALITY & FLEXIBILITY

Uniquely, using the aforementioned automatic link, you can output PERFECTION SE documents using over a thousand fonts (a huge variety of styles and sizes, supplied on the PUBLISHER and TOOLBOX disks) on virtually any printer – from the humblest Epson RX80, Brother M1009 or Star LC10 (which are all single font machines when used with most word processors) to top-end lasers. ***You are not limited to the fonts built into the printer!!*** All PERFECTION SE **bold**/underlined/ *italics*/$^{super}$/$_{sub}$, etc, settings are preserved. Proportional spacing and micro-justification are automatic, even when you mix fonts of differing widths ***and heights*** (even on the same line), vary line spacings, etc. Uniquely, you are not trapped with one type of micro-justification (ie adding all the space between words, and using the predefined widths of characters as their separation) – with our system, you can vary (in 5% steps) the proportion of micro-spaces added between words to that added between characters (the latter in proportion to their *individual* widths). Settings around 65%-35% – not the 100%-0% forced upon you by some other word processors – seem to give the most pleasing results. Uniquely, you are not limited to mere rectangular columns plus headers/footers – that's all the rest can do – you can output in any sequence to any number of frames (text flowing from one to the next), each of any shape – irregular polygons of up to 66 sides, circles, multi-column or part-column boxes (hundreds of types of borders, thousands of textures), doughnuts, wrap-around shapes, even re-entrant ones ('join-the-dots' type borders, even with intersecting edges) – all with micro-justification and proportional spacing! Look at the example on this page. Of course, if super fancy output or special effects are not of the essence, PERFECTION SE's direct printer output is more than capable of meeting your needs.

## THE FASTEST

For benchmarking, we've used an unimpeachable file – not one created specially – a public domain version of the first book of The King James Bible, all fifty chapters of the book of Genesis. This came to **one hundred and forty pages**, well over **forty two thousand words** excluding headers and footers, well over **two hundred and twelve thousand characters** excluding justification ones and **one thousand five hundred and thirty three indexed verses**!! We didn't use a smaller file (as used to benchmark other programs) as PERFECTION SE's timings for most operations then become impossible to stopwatch (too fast!). The hardware used for all timings was GOLD CARD: speeds would be **further improved by over three times** using the SUPER GC. Of course, LIGHTNING SE was used. File operations were to ramdisk: normal slave blocks would give identical times. All settings on **everything** were for maximum speed, except where indicated to the contrary – we have the sense **not** to force full speed upon you in operations like scrolling and global Search & Replace. PERFECTION SE's speed for these is switchable (at run-time and when configuring), as too great a speed may cause overshoot (with scrolling) or fatal alteration (if there is human error inputting the target or replace strings). Here are the benchmarks for this huge file:

Load 140 pages: 0.6 seconds (yes 0.6, not 6!) ☆ Import 140 pages: 0.6 seconds (yes 0.6, not 6!) ☆ Save 140 pages: 0.5 seconds (yes 0.5, not 5!) ☆ Export 140 pages: 0.5 seconds (yes 0.5, not 5!) ☆ Case-sensitive search from top for word at bottom: 0.4 seconds (yes 0.4, not 4!) ☆ The same, but case case-insensitive: 0.5 seconds (yes 0.5, not 5!) ☆ Case-sensitive search backwards from bottom for word at top: 0.4 seconds (yes 0.4, not 4!) ☆ The same, but case-insensitive: 0.5 seconds (yes 0.5, not 5!) ☆ Automatic Search & Replace, in Fast (No Query) mode, of last 600 occurrences: 7.4 seconds (same length replace string); 7.7 seconds (shorter replace string); 10.5 seconds (longer replace string – longer time as we deliberately chose a high *density* of replaces to handicap PERFECTION SE into auto-managing memory – without causing any heap fragmentation, but still with only a 0.005 second overhead per replace!). ☆ Automatic Search & Replace in Slow ('Querying') mode: arbitrarily slow, typically 30 times slower – because we deliberately allow for human response time (in case you want to abort) before proceeding from one replace to the next – booby prize to anyone for benchmarking us on this setting!! ☆ Scrolling 100 lines of text, up or down, by full-width screen page: 1.5 seconds ☆ Scrolling 100 lines of text on full-width screen, line by line, in slow (full) mode: 5.7 seconds (down)/5.8 seconds (up) ☆ As above, but in medium speed mode: 4 seconds ☆ The same, but in fast mode and default settings: 13.5 seconds to scroll through the whole massive document, averaging 0.23 seconds per 100 pages (!) – and this could be made up to ten times faster by reconfiguring PERFECTION SE ☆ Reformatting paragraphs, changing margins, justification, etc, of existing text: c5 times faster than predecessor ☆ Inserting (or undoing) emphasised, underlined, italics, superscript, subscript, 8 strips, 6 environment settings: Instant (i.e. immeasurable) ☆ Navigation to line or page or to top or bottom or to 8 markers or to highlights/blocks: Instant ☆ Setting new margins, justification, etc: Instant ☆ Deleting block of 100 pages: 0.3 (yes, 0.3 not 3!) seconds ☆ Copying/moving block of 100 pages (not just 10!), downwards or upwards: 3.4 seconds (yes, including all the time for automatic memory management and anti-fragmentation – other programs are light-years behind) ☆ Spellcheck as you type: Ten times faster than anyone can possibly type ☆ Spellcheck all 140 pages in the document using the 350,000 word Mega Dictionary: 3.9 seconds (20 'errors' – like 'pluckt'!) ☆ And using our tiny dictionary (well, tiny by our standards – large by comparison with most others): 5.1 seconds (566 'errors') ☆ Time taken to create user dictionary from the results of the second spellcheck (566 errors): 0.8 seconds to extract all 'errors' from document and clean document; 1.9 seconds to create a full user dictionary therefrom and also a sorted, duplicate-free wordlist file (for browsing) ☆ Spellcheck file (ASCII or native): Even faster. ☆ Print first 10 pages to file: 3.5 seconds. ☆ Change every occurrence of God to ***God*** in bold underlined italics, strip 2 – 9.5 seconds!

**TECHNICAL NOTES** Reformatting is the amendment of a section of previously-entered text to conform to margin, indentation, justification and pagination settings after the user returns to it and makes alterations, either by hand (by over-typing, deleting, adding or otherwise changing) or using search and replace, merge etc. PERFECTION SE lets the user pre-configure, or tune at run-time, the desired reformatting behaviour. The options are to either select Never (most suitable for technical users, and what all previous PERFECTIONs did: you had to initiate the reformat of the re-edited para), Instant (= 0.1 seconds, giving in-situ real-time automatic reformatting as-you-type: common in word processors, and irritating to the eye) or User-delay, the most flexible setting (giving slightly delayed auto-updating of lower text). On User-delay the user is free to set any delay from 0.2 seconds to 99.9 seconds in 0.1 second steps. About 1.5 seconds is best for sedate typists and 0.3 seconds for speed demons. This means that you are not hassled by continuing screen changes on lines below the one you are editing and concentrating upon, or shufflings around on the current line caused by right or centre justification. When you pause in your typing for longer than the set delay, PERFECTION SE automatically tidies up, without you having to do anything. On User-delay, if you navigate or progress off the line, or invoke any menu or command (including Save, Print etc.), an auto-reformat occurs instantly. This means that you are never left with the document in the wrong state. With these options, you have the best of all possible worlds.

Also, SHIFT/CAPS now obeys the indent margin (which matters if the cursor is on the first line of a para) and leaves the cursor position unaltered within the text. Other reformatting commands are unaltered, so you can still step through paras reformatting manually as you go, if you wish. The maximum number of lines, characters, words, lines, pages etc have all been increased (effectively to infinity: e.g., the new limit on characters is 30 million-million, this up from 2 million, restrictive in Super Gold Card / QXL days!). Also, the new version (starting with v5.13) is even faster, and its handling of complex search/replaces (say, involving end of line codes) has been optimised. PERFECTION SE really is superb!

## DIGITAL PRECISION PRICES

| | £ |
|---|---|
| 3-D PRECISION CAD SYSTEM | 49.95 d |
| ADVENTURE CREATION TOOL SPECIAL EDITION | 49.95 r |
| ARCADIA GAME | 9.95 s |
| ARCHIVE DEVELOPMENT SYS + RUN-TIME MODULE | 29.95 a |
| ARCHIVE TUTORIAL | 19.95 a |
| BETTER BASIC EXPERT SYSTEM | 24.95 a |
| BLOCKLANDS GAME | 9.95 s |
| CASH TRADER v3.3 + ANALYSER | 99.95 c |
| COMPARE FILE UTILITY | 19.95 a |
| COPY UTILITY FOR BACKING UP | 9.95 a |
| CPORT BASIC TO C AUTO-TRANSLATOR WITH C68 COMPILER | 99.95 c |
| CPORT BASIC TO C AUTO-TRANSLATOR | 89.95 c |
| DATABASE ANALYSER FOR ARCHIVE | 19.95 a |
| DAT-APPOINT APPOINTMENT DATABASE SYSTEM | 19.95 d |
| DESKJET/LASERJET DRIVER FOR PRO PUBLISHER | 19.95 c |
| DESKTOP PUBLISHER SPECIAL EDITION | 39.95 c |
| DESKTOP PUBLISHER | 24.95 d |
| DIGITAL C COMPILER | 29.95 a |
| DIGITAL C SPECIAL EDITION | 49.95 a |
| DISKTOOL WITH QUICKDISK | 19.95 b |
| DROIDZONE GAME | 9.95 a |
| DR-DOS v7 STATE OF THE ART PC OPERATING SYSTEM | 89.95 r |
| EDITOR SPECIAL EDITION | 49.95 d |
| EDITOR | 29.95 a |
| EYE-Q GRAPHICS SYSTEM | 39.95 a |
| FONT ENLARGER | 19.95 r |
| GAMES COMPENDIUM (ALL FIVE GAMES) | 29.95 a |
| GRAFIX 24-PIN UNIVERSAL PRINTER DRIVER | 14.95 r |
| HARDBACK + FINDER HARD DISK MANAGER | 49.95 c |
| IDIS INTELLIGENT DISASSEMBLER | 24.95 a |
| IDIS SPECIAL EDITION | 39.95 d |
| LIGHTNING ACCELERATOR | 24.95 a |
| LIGHTNING SPECIAL EDITION ACCELERATOR | 49.95 h |
| LIGHTNING GOLD SPECIAL EDITION | 39.95 a |
| MAILMERGE DATABASE SYSTEM | 19.95 a |
| MEDIA MANAGER SPECIAL EDITION | 49.95 d |
| MEDIA MANAGER | 24.95 a |
| MEGA DICTIONARY | 29.95 g |
| MICROBRIDGE | 39.95 d |
| NAMES AND ADDRESSES DATABASE SYSTEM | 19.95 d |
| PAYROLL SYSTEM | 49.95 c |
| PC CONQUEROR GOLD SPECIAL EDITION WITH DR-DOS v7 | 179.95 g |
| PC CONQUEROR GOLD SPECIAL EDITION | 99.95 g |
| PC CONQUEROR WITH DR-DOS v7 | 139.95 r |
| PC CONQUEROR | 59.95 r |
| PEDIT PRINTER DRIVER FOR XCHANGE | 19.95 a |
| PERFECTION PLUS SPECIAL EDITION WITH SPELLCHECKER | 139.95 c |
| PERFECTION PLUS WITH SPELLCHECKER | 99.95 r |
| PERFECTION SPECIAL EDITION WORD PROCESSOR v5 | 99.95 c |
| PERFECTION SPECIAL EDITION DEMO VERSION + SAMPLE DATA | 19.95 c |
| PERFECTION WORD PROCESSOR v3 | 59.95 d |
| PERFECT POINTER TOOLS | 29.95 a |
| PROFESSIONAL ASTROLOGER WITH ASTRONOMER | 69.95 a |
| PROFESSIONAL ASTROLOGER | 59.95 a |
| PROFESSIONAL ASTRONOMER | 29.95 s |
| PROFESSIONAL PUBLISHER TOOLBOXES (BOTH PARTS) | 49.95 r |
| PROFESSIONAL PUBLISHER TOOLBOX PART ONE | 29.95 r |
| PROFESSIONAL PUBLISHER TOOLBOX PART TWO | 29.95 r |
| PROFESSIONAL PUBLISHER | 89.95 c |
| QFLICK CARD INDEX SYSTEM | 29.95 a |
| QKICK FRONT END SYSTEM | 24.95 a |
| QMATHS MATHEMATICAL SYSTEM PART ONE | 69.95 c |
| QMATHS MATHEMATICAL SYSTEM PART TWO | 59.95 c |
| QMATHS MATHEMATICAL SYSTEM (BOTH PARTS) | 99.95 c |
| QMON MACHINE CODE MONITOR | 39.95 a |
| QUICKLASER PUBLISHER DRIVER FOR LASERJET/DESKJET PRINTERS | 19.95 c |
| RECOVER ARCHIVE UTILITY | 19.95 a |
| REVERSI GAME | 9.95 s |
| SEDIT + SCREENPRINT ARCHIVE UTILITIES | 19.95 a |
| SOLUTION PC EMULATOR WITH DR-DOS v7 | 109.95 r |
| SOLUTION PC EMULATOR | 29.95 r |
| SPELLCHECKER WITH DICTIONARIES | 49.95 d |
| SUCCESS CP/M EMULATOR | 49.95 b |
| SUPERBASIC MONITOR | 24.95 a |
| SUPERFORTH COMPILER WITH REVERSI | 39.95 a |
| SUPER ASTROLOGER | 24.95 s |
| SUPER BACKGAMMON GAME | 9.95 s |
| SUPER SPRITE GENERATOR | 29.95 a |
| TOOLKIT III WITH ROM | 49.95 h |
| TOOLKIT III | 29.95 a |
| TRANSFER UTILITY SPECIAL EDITION | 29.95 b |
| XREF SUPERBASIC PROGRAM ANALYSER | 29.95 a |
| TRANSFER UTILITY | 9.95 b |
| TURBO BASIC COMPILER + TOOLKIT | 79.95 a |
| ULTRAPRINT SCREEN DUMP UTILITY | 19.95 a |
| XREF SUPERBASIC PROGRAM ANALYSER | 29.95 a |

KEY a=Available either on cartridge or disk; b=Disk only; c=Minimum 512K exp: disk only; d=Minimum 256K exp: either cartridge or disk; r=Minimum 256K exp: disk only; s=Cartridge only; g=Minimum 1.5Mb RAM: disk only disk; h=ROM + (cartridge or disk)

### AND NOW - BEST-EVER SPECIAL DEALS...

5% off 2 programs
10% off 3 programs
20% off 4 programs
30% off 5 programs
40% off 6+ programs!

# SUPER GOLD CARD

This amazing product is the way forward for the QL. Like Gold Card before it, the brand-new **Super Gold Card** is a plug-in disk interface and RAM expansion that works on all QL versions. Incredibly, it is *over three times* speedier than Gold Card, with *over twice* the RAM and with many enhancements. It complements LIGHTNING SPECIAL EDITION like nothing else, squeezes the best out of TURBO (which was designed with 32-bit CPUs in mind) and really accelerates PC CONQUEROR. **Super Gold Card** is actually as fast, or slightly faster, than the much vaunted QXL: also, it is **100%** QL-compatible **now**, and no PC is needed. The table below really says it all:

| *System →* / *↓ Features* | *Bare QL* | *TRUMP CARD I* | *GOLD CARD* | **SUPER GOLD CARD** |
|---|---|---|---|---|
| Relative Speed | 1x | 1.8x | 7x | **25x !** |
| Motorola CPU | 68008 | 68008 | 68000 | **M68020** |
| Clock Frequency | 7.5MHz | 7.5MHz | 16MHz | **24MHz** |
| Bus width | 8 bit | 8 bit | 16 bit | **32 bit** |
| RAM fitted | 128Kb | 896Kb | 1,920Kb | **3,968Kb** |
| RAM access speed | Slow | OK | Fast | **Twice as fast** |
| PCB population | V.high | High | V.low | **V.low** |
| Physical dimensions | Monolith | Full-size | Half-size | **Half-size** |
| Lock-up frequency | Ouch! | Occasional | V.rare | **Won't** |
| Battery Backup Clock | No | No | Yes | **Yes** |
| Clock Protection level | N/A | N/A | Modest | **High** |
| Toolkit II + Manual | No | Yes (early vns) | Yes | **Enlarged** |
| Sub-directory support | No | No | Yes | **Yes** |
| Parallel/Centronics port | No | No | No | **Yes** |
| Spooler/Screendump/Ramdisks | No | Yes | Yes | **Yes** |
| Speedup switch (Screen#2) | No | No | No | **Yes** |
| Future hi-res graphics | No | No | No | **Planned** |
| Disk drives supported | N/A | SD/DD | SD/DD/HD/ED | **SD/DD/HD/ED** |
| Max no: of disk drives | 0 | 2 | 3 | **4** |
| Max sectors/disk | N/A | 1,440 | 6,400 | **6,400** |
| Max disk transfer rate | N/A | 30Kb/sec | 120Kb/sec | **>120Kb/sec** |
| Peripheral card tolerance | OK | No | No | **OK** |
| SCSI-2 compatibility to-be | No | ? | No | **Yes** |
| Miracle/DP Warranty | No | No | 2 years | **2 years** |
| DIY/Kit incorporability | Yes | No | No | **Yes** |
| Overall Rating by DP | 2% | 10% | 30% | **110%** |

**And to the Very Best news: from DP, SUPER GOLD comes SUPER CHEAP!**
**SUPER GOLD CARD, plus a no-limit extra 20% SOFTWARE DP DISCOUNT VOUCHER, plus a FREE mystery DP program, plus a FREE Dust Cover, will cost you a mere £375 ✓✓✓✓**
**Less £125 if part-exchanging your standard 2Mb Gold Card**
**Add £125 for ED 6400-sector Disk Drive (PSU, cased, cables).**

OTHER HARDWARE EXCHANGED AND SOLD BY ARRANGEMENT. PLEASE ORDER NOW: WE EXPECT VERY HIGH DEMAND FOR SUPER GOLD CARDS, AND WE DON'T WISH TO DISAPPOINT. INTERNATIONAL RAM PRICES ARE UNSTABLE AND PRICE HIKES MAY BE INEVITABLE. CONSEQUENTLY, THE ABOVE COMBINATION OFFER COULD BE WITHDRAWN WITHOUT NOTICE...

**The software discount must be taken at the same time as the main order, and CAN be combined with the SPECIAL DEALS discounts. For example, if you chose six DP programs of total list price £100, you would only have to pay £100 -40% -20% = £48 for them! And, of course, you would also get the two gifts absolutely free, and a 4Mb SUPER GOLD CARD too!**

*TERMS & CONDITIONS * We accept payment by VISA, Mastercard, Access, Eurocard, Eurocheque, £ cheque drawn on a UK branch of any bank or building society, postal order, cash, travellers cheques or direct funds transfer to our bank account #60327808 at Barclays Bank plc, South Chingford Branch (sort code 20-53-00), NE London Business Centre, PO Box 2403, London N18 2BY. If you must use another currency, add 7.5%. If you must use another form of cheque, add £15. * Upgrades cost only the difference in current price plus £10; return the original disk and manual. But do NOT return the manual if the upgrade is to a "SPECIAL EDITION" of your original program: you will be sent useful supplementary documentation. Upgrades from LIGHTNING are the exceptions to the last rule! * Only DP programs and upgrades (i.e. not DR-DOS) count towards discounts. * Prices are all-inclusive for UK mail-order. Rest of Europe, add 5%. Elsewhere, add 10% (Air Mail). * We aim for 24 hour turnaround. * DP & Digital Precision are trading names of Digital Precision Ltd, Company No.1833989, registered in England & Wales. * Orders using your own paper are also welcome!*

Post to: *DIGITAL PRECISION*, 222 The Avenue, London E4 9SE, UK
or phone on 081-527 5493 from UK, +44 81-527 5493 rest of the world

Name:_____ Address:_____

Products ordered: _____

on mdv/3.5"disk/5.25"disk* (no: of sectors on disk-format = 360/720/1440/2880/6400*; if 5.25"/720, specify 40/80* track) **delete as needed.*

Signature: _____ Amount: £_____

Card No: _____ Expiry date:_____

VOL. 3 ISSUE 3

# QL Contents

*Editor*
Helen Armstrong

*Publisher*
Mark Kasprowicz

*Advertising Manager*
Jim Peskett

*Creative Director*
John Stanley

*Graphic Artist*
Stevie Billington

*Magazine Services*
Linda Miller, Frances Maxwell,
Pauline Wakeling,
Val Brandon

Sinclair QL World,
Published by Arcwind Ltd.
The Blue Barn,
Tew Lane, Wootton,
Woodstock,
Oxon. OX7 1HA
Tel: 0993 811181
Fax: 0993 811481
ISSN 026806X

If you have any comments or difficulties please write to the editor and we will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.
Back issues are available from the publisher price £2.50 UK, £2.99 Europe. Overseas rates on request.

Subscriptions from: Arcwind
The Blue Barn, Tew Lane,
Wootton, Woodstock, Oxon.
OX7 1HA
UK: £23.40
Europe: £32.90
Rest of World: £40.90

Reprographic Services: Eclipse, Brook Street, Watlington, Oxon. OX9 5JH.
Distributed by: Seymour Press Ltd., Windsor House, 1270 London Road, Norbury, London, SW16 4DH

# QL Scene

## Di-Ren Plan Whole Interface Range

Di-Ren have altered theirprojected DPR020 Sound Analyser. Instead of an add-on, it is to be incorporated into **System Amadeus**, a new I/O comprising one controller device into which all other devices are plugged. "Unfortunately," says Robin Barker, "We could not achieve an acceptable level of sound quality at normal QL RS232 serial port baud rates. We considered fitting on-board memory, but this would add significantly to the cost.

Di-Ren's advance information on the Amadeus data transmission system - expected in April 1994 - describes a device which will connect any number of I/O interface cards (such as Centronics printer ports) to the computer, connect terminals with like or unlike terminasl, and connect terminals with interface cards.

Three types of **controller** are at the heart of Amadeus: **Centronics** controller, **RS232** configures a DTE, capable of communicating at up to 115000 baud, and **QL rom port** controller. The controllers are accompanied by the appropriate software, currently for QLs and PCs. The Controllers, Centronics interface and Sound interface are likely to be the first releases, and Di-Ren would like to hear from the public what else is in demand. Network driver software is also on the "coming soon" list.

A sound interface and a data buffer intoface are also on the list, as is a power adaptor which can be plugged in-line to accept a power range of 7-20 volts AC or DC, and a power enhancer for long cable runs.

Tentative prices show the Controllers and I/O Interfaces in the £27-£39 price range. Other prices are still TBA. The system is still under development and advancing by the minute. For the most up to date leaflets or other queries, contact **Di-Ren at 59 William St., Walsall, West Midlands WS4 2AX. Tel. 0922 33580, or Dilwyn Jones,** who will be the distributors.

Meanwhile, Di-Ren's **QL-PC Fileserver** will shortly be upgraded to SE standard, with optional link port facilities and improved coding. This will be obtainable as an upgrade package for current users.

A new program, QL-PS Superlink, will allow communications and complete control of a QL via a PC, using the Amadeus system under many circumstances. This program will be "complex and a little time coming", say Di-Ren.

## Surrey Quanta

***Quanta are holding a Surrey Workshop in Byfleet Village Hall, High Road, Byfleet on May 14th 1994. Contact Ken Bain on 0923 347432 for more details. It has been pointed out that this date clashes with the IQLR Show in Newport, Rhode Island, USA on the same day. The complications of an international show timetable!***

## MERZ UPDATES

Jochen Merz's **QVME** QL emulator (now for Atari Mega STE and Atari TT) now costs only DM 549. It supports all three serial ports on the Mega STE, with split baud rates and buffering on all three. All support rates up to 19200 baud, and the third one also supports 38400 and 57600 baud. The emulator's boot software now boots directly from every hard disk. The current verions is Level E41, and the manual is Revision 8.1.

**QD version 6** has added the important feature of automatic tabulation character compression and decompression. You can choose which files can be tab compressed. This means that the file can be accurately reproduced when re-loaded, but can save between 25 and 40% of file space on saving (assembler sources). **Compression** is a little slower (about 10%) than before, but loading time is faster. Assembly time is also saved as the assembler only has to skip tabs, not multiple spaces. Check whether your assembler will accept tabs before compression (GST Macro Assembler, for instance, will not).

QD now also has row and column display without slowing character entry down, even on slow systems. It will also check bracket match, in both directions, on all sorts of brackets and string defintors << and >>. There are other improvements as well.

For those ligher moments, Merz has another game - **DoubleBlock, a Tetris-like game** with good graphics, several difficulty levels and a rare player-vs-player option.

For information and news, call Jochen's **Qdos Box Duisberg on Germany 0203 591706**, 24 hours, no user charge. The box supports currently up to 16800 baud for high speed modems to save on phone bills for overseas callers. As well as product news and version numbers, the box has PD software on it, including demo versions of Jochen's and other software, free to download.

Qdos Box **File areas** include: Mailbox utilities, Assembler, SuperBasic, C and Pascal, Text87, General Utilities, Other Software (such as Qdos Xchange), Demos, General Documents, Pictures (GIF and UNGIF file viewer) and others. Menus are in English and German.

**Jochen Merz Software, tel. 0203 501274, mailbox 0203 591706.**

# QL Scene

## Page Designer 3 and File Master

Dilwyn Jones has been shelling out press releases at a furious rate this month. **Page Designer 3** version 1.09 is now out incorporating suggestions from the first crop of PD3 users. There is now up to x5 zoom available while editing, page colour inversion, an Overstrike option for multiple-pass printing, and a considerable number of bug-fixes.

Current users who want the upgrade should return their master disk to DJC with return postage for a free upgrade to the current version. PD3 normally costs £40, or £25 to upgraders from PD2 with proof of purchase.

Dilwyn is requesting that anyone who may have sent for information, or an uncashed order, well in advance of release and has not heard from DJC, please to contact him so that they can set the records straight.

Also now in upgrade is Joe Haftke's **Filemaster**, now catering for the big increases in QL memory expansion and disk storage capacity.

Filemaster can now manipulate selections up to 288 files (subject to free memory). Meanwhile, Directory functions from DIR MENU have no restrictions on the number of files and are only limited by free memory. Provided sutiable memory is available, Filemaster_Plus will run with Trump Card QLs as well as Gold Card.

The **Filemaster User Guide** has been updated to the new system. FIlemaster costs £12, or for existing users to upgrade, £2 (UK) or £3 (overseas) with return of the master disk.

Contact **Dilwyn Jones Computing, 41 Bro Emrys, Tal-Y-Bont, Bangor, Gwynedd, LL57 3YT. Tel. 0248 354023.**

## Quanta 10th Anniversary Show

Quanta are publicising their **10th Anniversary AGM and Workshop** with more verve than usual and clearly hope for a record turnout of and visitors.

The Meeting is to take place at the Park Hotel, Tynemouth, on Saturday and Sunday 23-24th April 1994. Tynemouth is a seafront town a few miles east of **Newcastle Upon Tyne** and near to road, rail and air via Newcastle Airport.TheStevenson Railway Museum nearby is another attraction.

On the QL front, the meeting is expecting demonstrations of the new **Super Gold Card** from Miracle Systems, and Ron Dunnett's new Fastnet and 16-bit IDE interface for the QL. The main Workshop, and the Anniversary dinner open to members and quests on a first-come first-served basis, are on the 23rd, and the AGM on the 24th.

The Park Hotel, Grand Parade, Tynemouth can be contacted on 091 257 1406 (overseas +44 91 257 1406) or fax 091 257 1716. There are 400 parking spaces. Information from the local organisers should be directed to **Derek Stewart, 20 Emily St., Gateshead, Tyne & Wear, NE8 3QH.**

## Memory Watch

**WatchMem_obj** is a utility job allowing assembler and C programmers to monitor an area of memory and to give the alarm and a message when that memory is altered. Written by Ian Bruntlett, find it at **Quanta, Qubbesoft P/D and Jochen Merz's Qdos mailbox.**

## Miracle in Newport

IQLR's Miracle In Newport **show,** which held its first meeting last year sponsored by Miracle Systems, will take place as planned on Saturday 14th May in Newport, Rhode Island, USA (near New York).

This year the show will be held at the **Meeting Room, Howard Johnson Lodge, 351 West Main Road, Middletown, RI 02849, USA** from 10am to 6pm. (The Salvation Army Hall is not available this year.)

The show however is just the centrepiece of an extended weekend of meeting and socialising between US QLers and their visitors. Following last year's revels, this year will again see a Dutch Treat Dinner (the US version of a friends' night out where everyone pays their own dinner bill, well known to regular attendees of European QL shows) on the Saturday, a Brunch of coffee and doughnuts at Motel 6 from 9am on Sunday morning, and plenty of other opportunities to socialise for those who arrive early and stay late.

Visiting UK suppliers will be staying at **Motel 6, 249 Connel Highway, Newport**. A single room costs US$38.07 per day at the motel, and international reservations can be contacted on +1 505 892 8667. But, reports IQLR, there are plenty of other motels and hotels in the area for anyone who turns up unexpectedly.

From Europe, the show is expecting to see Stuart Honeyball of Miracle, Bill Richardson of W N Richarson, Dilwyn Jones of DJC and his wife Janet, and Ron Dunnett of Qubbesoft, plus many representatives of US suppliers and newsletters.

**Contact Bob Dyl on +1 401 849 3805 for more information.**

# OPEN CHANNEL

## QED Search

Risking accusations of looking a gift-horse in the mouth, could I plead with Mr. Bredenbeek, when he next looks at his excellent editor **QED**, in the C68 PD, to amend the search facility 'F', so that the search string is not cleared until the leaving of the editor, or until a new string is entered as in Metacomco's ED?

I would apologise to you all for the scrawl, but it's really all your fault. Every time I think of buying a **printer**, Troubleshooter publishes another printer article that puts the fear of God into me.

**G F Fisher**
**Bristol**

*Come now! No computer system is complete without a printer. Let's have some letters of recommendation, please, for recently-purchased, well-behaved, reasonably-priced QL-compatible printers from any dealers who offer a helpful level of support for a new user. They are a pain at times, but where would we be without them? Knee-deep in discarded biros.*

## Index Unbound

Now that Mike Lloyd has finished his **Keyword Index**, I feel he should be congratulated on the work he has put into the project. With hindsight, the index provides a very valuable contribution to QL reference sources, bringing together and documenting, as it does, all the major keywords which have been produced from the beginning.

Being distributed over 30 or so QL Worlds, however, detracts from its usefulness, since it is fairly difficult to slog through back issues to home into a particular keyword. Might I suggest that you consider reprinting this index as a stand-alone publication so that it can stand on the bookshelves in its own right?

Reprinting the index in this way would allow correction of the editing errors that have crept in. I make this comment in the nicest possible way - I am sure you would appreciate what I mean!

**Peter Tyler**
**Ormskirk**
**Lancs.**

*This is something we shall have to talk to Mike about. There are corrections we could make, but it would be quite a big project to update and publish the whole thing, as we have no existing page files where we can slot corrections in.*

*Of course, the centre-page layouts and "holes" down the middle mean that you can extract and re-bind your New User Guide as a separate unit without tearing your QL World backissues - the whole series was designed as a pull-out, after all! They should also be in alphabetical order, but we have to admit that something went slightly adrift around part 19.*

## QPAC Puzzle

This letter is prompted by the letter from my namesake, Roy Wood in QL World III.1 ("Wrong Definition") and his 20th rewrite of his **QPac2 boot** program. After a few attempts using the manual - largelly unintelligible - and after following through the QPacer prog and various articles, I produced a workable boot program.

My wish, after installing Miracle's Gold Card, was for a word-processor, Archive, Abacus, and QL_Genealogist operating together. My relief at achieving this has been dimmed slightly because my wordprocessor and QPac2 do not live together as completely as do either the Psion Suite or QL_Genealogist, although I can use program switching with **Taskmaster**.

The problem with **QPac2** is that under certain conditions, such as after using one of the Psion suite, trying to use F2 from Perfection to check a Menu Command results in my system locking up. I asked a member of the DP team about this at the Blackrod workshop, and while he had not heard of the problem, he did advise that one could use F1 and the Help screens instead of the Menu display. This is an alternative, but it is not as convenient or explicit as the F2 path. I wonder if any of your readers have had the same thing happen in QPac2, and if they have found a solution.

**Ernest Wood**
**Southport**
**Lancs.**

## How To Ramdisk

Arthur Nunn wrote in Open Channel in QL World III.1 that he knew little about ramdisks. As a trader, I have found this surprisingly common, even though there is information about them in all of the modern disk interface manuals. To avoid lengthy phone calls trying to explain things like this, I produced some information sheets for customers some time ago. There are three in all, one on ramdisks, one on the **Pointer Environment**, and one on **printer** problems (Aaargh! I'm sure you know all about those!). You can use them in the magazine if you require to, though bear in mind that bits have appeared in other QL publications.

Alternatively, these are available to customers and readers free from DJC

**Open Channel is where you have the opportunity to voice your opinions in Sinclair QL World. Whether you want to ask for help with a technical problem, provide somebody with an answer, or just sound off about something which bothers you, write to: Open Channel, QL World, The Blue Barn, Tew Lane, Wootton, Woodstock OX7 1HA.**

with orders on request and with catalogues. Otherwise, please send return postage. We can also supply a ramdisk on disk or cartridge for £2 (UK) or £3 (overseas).

**Dilwyn Jones**
**DJC**
**Bangor**

## More Modems

Would it be possible for there to be in the not too distant future an article covering the topic of **modems and electronic communications** in relation to the QL? Like many readers I own a Tandata QL modem but now find that its slow speeds preclude access to many networks (not least my University's mainframe). The problem created by this situation is (a) what should I replace the Qmod with (b) how does one physically connect a non-QL designed modem to the QL (c) what software can/should be used (d) what are some of the bulletin boards available, UK and internationally? In brief, what is the current and likely situation vis-a-vis the QL in this important field

Best wishes for the New Year.

**Dr. Neil Daglish**
**Porirua**
**New Zealand**

*We did an article on Bulletin Boards in II.8 (often, incorrectly, known as II.7, as opposed to II.7, which is also, quite correctly, known as II.7). We have been talking about modems recently and will talk further.*

## NOT FOUND

Hopefully this may help some readers: for some time I have been losing disk data and programs with the dreaded NOT FOUND error message. Sometimes on one or other of the drives, sometimes on both with no obvious pattern. I then remembered a QL article a long time ago which commented that sometimes the QL processes information faster than peripherals can deal with it. I now place a PAUSE 200 after the "delete drive 1" instruction, another after saving and a third after deleting drive 2. Bingo! No more error messages.

**Don Smith**
**Kirkbymoorside**
**Yorks.**

*Thanks, Don.*

## Poke Puzzle

I would like to thank Bruno Coativy for the letter in Open Channel Vol II.12 (Poke modification). Unfortunately his last suggestion of how to poke in the protected area of the Gold Card is wrong. I tried also a simple machine code program to modify original Font 2 without any positive effect. My conclusion is that the protection of the rom area of the Gold Card is very sophisticaed, and without precise knowledge of this matter any of the experiments are hopeless.

It would be very nice if I can get the address of Mr. Coativy for direct contact.

Keep up the good work on QL World and thanks for this magazine. Only the delays in issuing are not so pleasant.

**Dr. Vladislav Teply**
**Cilova 89**
**162 00 Prague 6**
**Czech Republic**

*The words "no kidding" spring to mind. Just when I thought that the only things that could hold us up last month were under my control, a monitor broke down and then there was a snowstorm and somebody's advert went round the long way. Perhaps if I stop worrying, we'll be early!*

*I will forward your letter to Bruno. You could collaborate and perhaps approach Miracle too.*

## Locals/Parameters

In **SuperBasic in Action**, Vol II.7, Simon Goodwin comments about the "nine locals/parameters" problem. In fact, it may be a dramatic problem if you really need more than nine parameters into a recursive procedure or function. But the problem occurs only if you declare too many locals/parameters in the SAME procedure. So, if your procedure BIG_WORK needs, say, 15 locals (var1,...,var15), it's possible to survive by writing something like:

```
100 DEFine PROCedure
BIG_WORK
110 LOCAL var1 ... var9
120 BIG_WORK_a
130 END DEFine
BIG_WORK
140 DEFine PROCedure
BIG_WORK
150 LOCAL var10 ... var15
160  and here is your
170    code
180 END DEFine
BIG_WORK_a
```

Of course, if you need more than 18 locals you will use BIG_WORK_b, BIG_WORK_c, etc. I can't confirm that this method always works with every QL's configuration, but as far as I am concerned I never had had any trouble with it.

Long life to the QL and Sinclair QL World!

**J Piette**
**Leige**
**Belgium**

*Simon says: "This works with locals, on all Qdos systems, but still limits the user to nine parameters as opposed to locals unless Minerva or compiled Basic is used."*

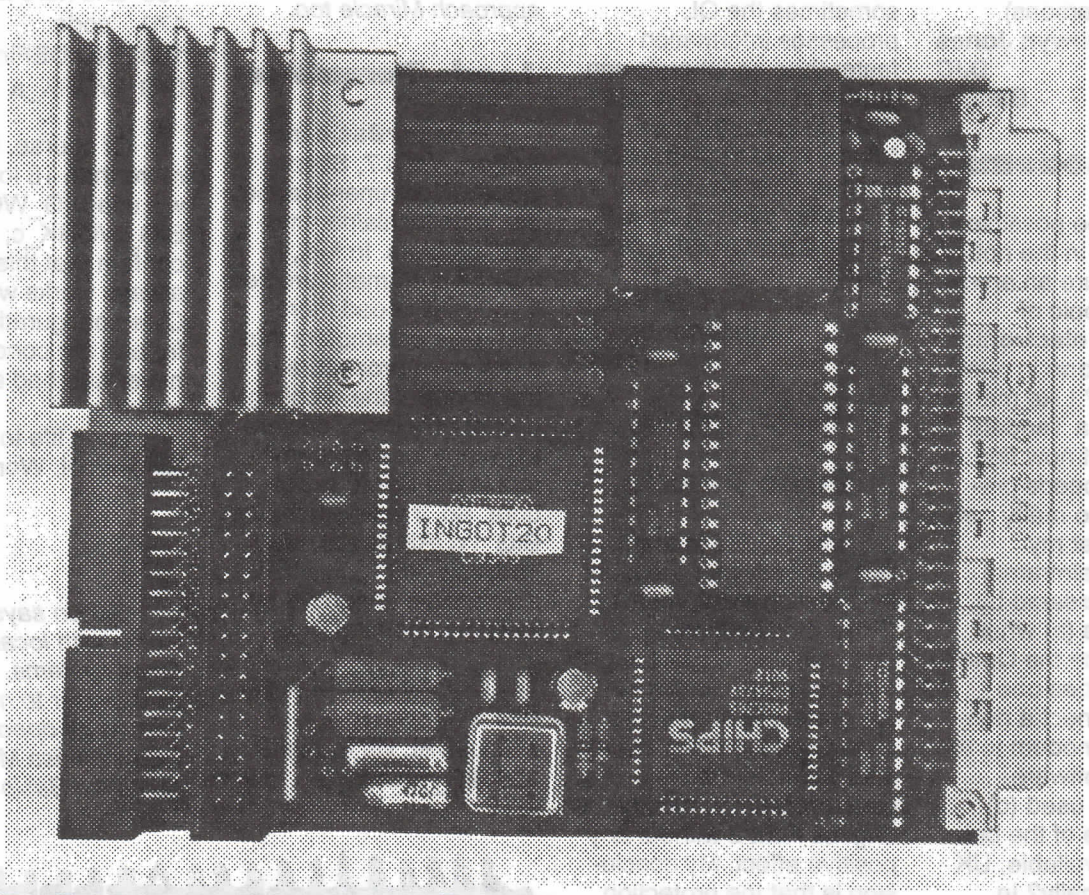
## Editor's Notebook

The accent this month is on file handling. SuperBasic In Action is the first part of Simon's development of Browser, a simple file requester. A short SuperBasic program by Neil Gordon (one of our most patient programmers!) helps with making backup disks. The New User Guide looks at the memory structure of Qdos Hilary Snaden, on the other hand, gives us one way of taking over the PC World.

Talking of "other" computers - how about a network which won't net, a computer which declines to load your applications, loses your files, refuses to respond to your investigations, and won't print because it's out of memory? You don't need a QL - no names, but you can pay eight times as much for all this!

The Royal National Institute for the Blind say that they have plenty of volunteers to help their members with computing now, thanks to everyone.

# MIRACLE SYSTEMS




# SUPER GOLD CARD

**Briefly...**

* 3 times faster
* 4M bytes of RAM
* CENTRONICS port

£375 inc.(£325 outside EU)
or GOLD CARD + £225
(£200 outside EU)

**What is it?**

This is the first major revision of our highly successful GOLD CARD. We have replaced the 68000 with a 68020 so programs run about 3 times faster and have expanded the memory to 4M bytes. Other improvements include a fast CENTRONICS printer port, 2 disk drive ports, virtually crash-proof clock and a socket to optionally connect 5V.

**The deal...**

The price is £375 inc VAT, (£325 outside EU) which covers postage, 2 year warranty and 14 day money back guarantee. We can upgrade your GOLD CARD for £225 (£200 outside EU). Alternatively, trade in your TRUMP CARD (£50), Super Q Board (£40) or other expansion (£25) against the purchase price Additionally, you can trade in your QL CENTRONICS (£15) or DISK ADAPTER (£10) or both

**To Order:**

Please make cheques payable to MIRACLE SYSTEMS LTD in pounds sterling or quote credit card number and expiry date.

MIRACLE SYSTEMS LTD
25 BROUGHTON WAY
YORK
YO1 3BG
U.K.
Tel: 0904 423986

**Why not subscribe?**

I Q L R

Contact
IQLR
23 Ben Curley Drive
Thetford
IP24 1QJ
Tel: 0842 762406

I Q L R

RECYCLED items with 1 year warranty

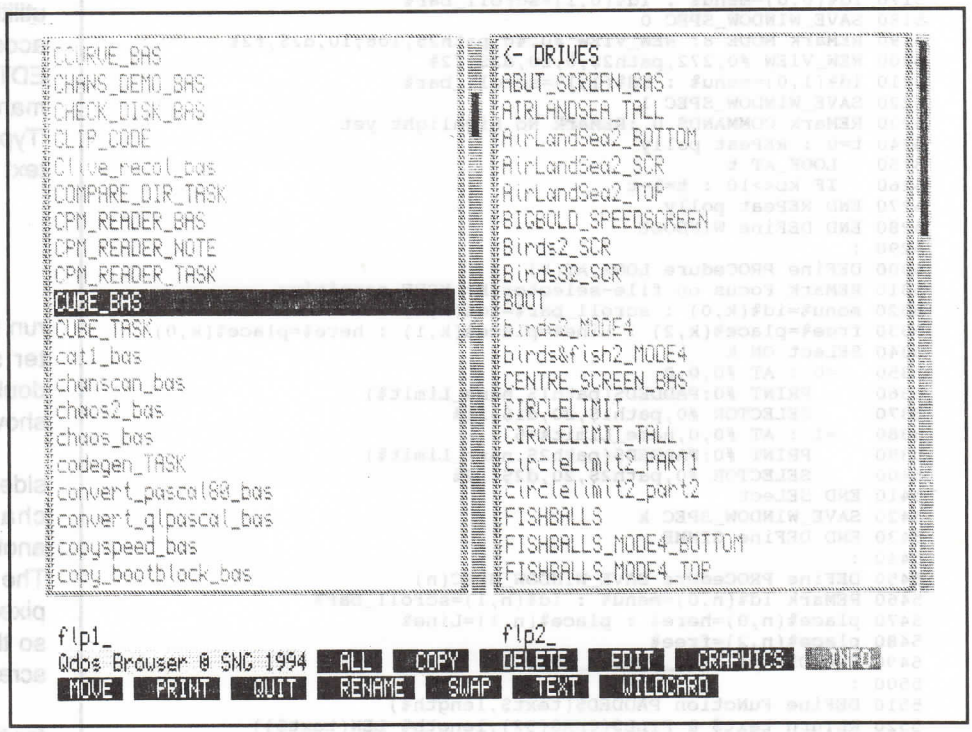
| | |
|---|---|
| GOLD CARD | £150 |
| QL CENTRONICS | £15 |
| DISK ADAPTER | £10 |

# SuperBasic In Action

**Simon N Goodwin goes browsing with his new SuperBasic project.**

This project is a logical application for the SuperBasic file requester developed last year. The Browser is a friendly, simple 'shell' for Qdos, allowing file manipulation, viewing and copying, without the need for SuperBasic commands.

The Browser lets you select files by pointing at their names on the screen. You can manipulate individual files as you go along, or mark a group - perhaps including sub-directories - for automatic processing. It works like DW and SID on the Amiga, or XTree and File Manager for PCs. It has two directory windows so you can see things move around as you work, it's easily extended and configurable for all sorts of display.

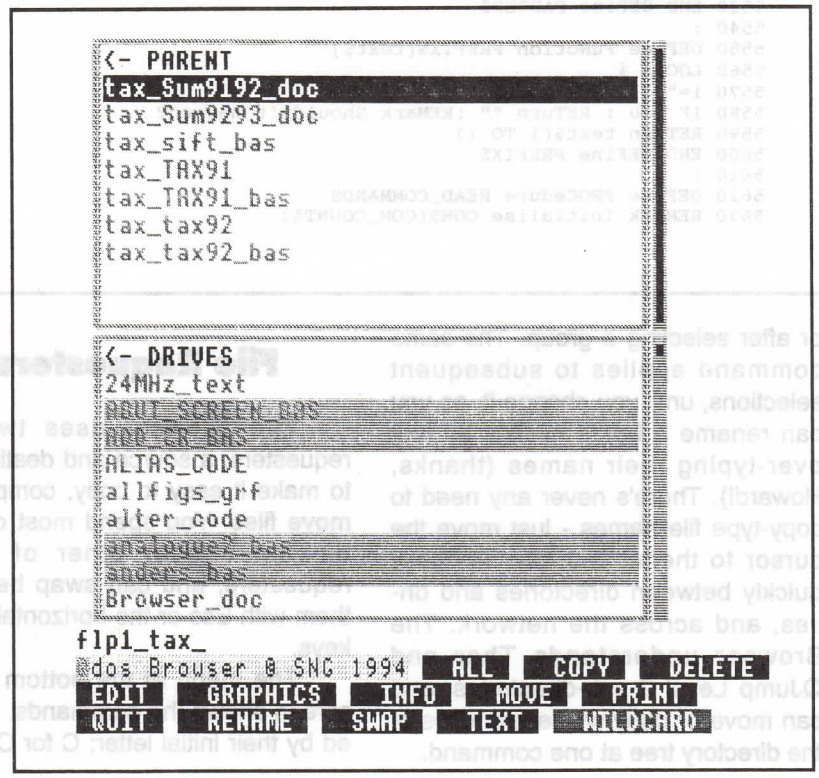



## Getting Started

What do you do first, when you get a new disk? Read the manual? Perish the thought. LRUN BOOT? Not if you expect a safe return to your familiar system thereafter. If you're anything like the QL owners I know, you whack it in FLP1_ and type DIR, or WSTAT, to find out what's there.

The next step depends on your style, but you'll probably look at a few files with COPY FILE,SCR (or MORE, if you have it) and maybe hop into an editor, using DIR again every so often for inspiration. Once boredom or paranoia set in, you may even make a backup with WCOPY or a utility like TCOPY. This is all good typing practice, but there must be an easier way...

This SuperBasic utility makes browsing through files much less tortuous. Commands are single letters and can apply to several files at once. You can issue them before you pick a file,



```
5000 REMark Qdos WINDOZE - SuperBASIC in Action April 1994
5010 REMark Uses FILE REQUESTER routines introduced last year
5020 REMark Copyright 1993,1994 Simon N Goodwin, version 0.4
5030 :
5040 DEFine PROCedure WINDOZE(n)
5050 LOCal i,t,com_code%
5060 REMark Windows are numbered from 0 to n-1
5070 READ_COMMANDS : com_code%=0
5080 DIM d1$(max_files%,name_Limit%),d2$(max_files%,name_Limit%)
5090 DIM f1%(max_files%),f2%(max_files%)
5100 REMark D2, F2 etc. for more windows (or use Minerva slices)
5110 DIM path1$(name_Limit%),path2$(name_Limit%)
5120 DIM x$(50),y$(50),wild$(name_Limit%),dest$(50)
5130 DIM id%(n-1,1),place%(n-1,2)
5140 path1$="flp1_tax_" : path2$="flp2_"
5150 REMark MODE 8: NEW_VIEW #0,40,path1$,0,10,d1$,f1%
5160 NEW_VIEW #0,24,path1$,0,20,d1$,f1%
5170 id%(0,0)=menu% : id%(0,1)=scroll_bar%
5180 SAVE_WINDOW_SPEC 0
5190 REMark MODE 8: NEW_VIEW #0,40,path2$,108,10,d2$,f2%
5200 NEW_VIEW #0,272,path2$,0,20,d2$,f2%
5210 id%(1,0)=menu% : id%(1,1)=scroll_bar%
5220 SAVE_WINDOW_SPEC 1
5230 REMark COMMANDS 0 :REMark No highlight yet
5240 t=0 : REPeat polly
5250   LOOK_AT t
5260   IF kb<>10 : t=1-t
5270 END REPeat polly
5280 END DEFine WINDOZE
5290 :
5300 DEFine PROCedure LOOK_AT(k)
5310 REMark Focus on file-selector K; MODE sensitive
5320 menu%=id%(k,0) : scroll_bar%=id%(k,1)
5330 free%=place%(k,2) : Line%=place%(k,1) : here%=place%(k,0)
5340 SELect ON k
5350   =0 : AT #0,0,0
5360     PRINT #0;PADDED$(path1$,name_Limit%)
5370     SELECTOR #0,path1$,20,d1$,f1%
5380   =1 : AT #0,0,name_Limit%+5
5390     PRINT #0;PADDED$(path2$,name_Limit%)
5400     SELECTOR #0,path2$,20,d2$,f2%
5410 END SELect
5420 SAVE_WINDOW_SPEC k
5430 END DEFine GIMME
5440 :
5450 DEFine PROCedure SAVE_WINDOW_SPEC(n)
5460 REMark id%(n,0)=menu% : id%(n,1)=scroll_bar%
5470 place%(n,0)=here% : place%(n,1)=Line%
5480 place%(n,2)=free%
5490 END DEFine SAVE_WINDOW_SPEC
5500 :
5510 DEFine FuNction PADDED$(text$,length%)
5520 RETurn text$ & FILL$(CHR$(32),length%-LEN(text$))
5530 END DEFine PADDED$
5540 :
5550 DEFine FuNction PREFIX$(text$)
5560 LOCal i
5570 i="_" INSTR text$
5580 IF i=0 : RETurn "" :REMark Shouldn't happen?
5590 RETurn text$(1 TO i)
5600 END DEFine PREFIX$
5610 :
5620 DEFine PROCedure READ_COMMANDS
5630 REMark Initialise COM$(COM_COUNT%)
```

or after selecting a group. The same command applies to subsequent selections, until you change it, so you can rename a batch of files by just over-typing their names (thanks, Howard!). There's never any need to copy-type file names - just move the cursor to them. You can navigate quickly between directories and drives, and across the network. The Browser understands Thor and QJump Level 2 sub-directories, and can move or copy whole branches of the directory tree at one command.

## File Requesters

The Browser uses two file requesters, a source and destination, to make it easy to copy, compare or move files. You spend most of your time in one or other of these requesters, and can swap between them with Esc or the horizontal arrow keys.

The menu at the bottom of the screen shows the commands, selected by their initial letter: C for Copy, T for Type (text) and so on. These commands correspond to SuperBasic subroutines. Add the name, and a matching subroutine, and you've taught Qdos Browser a new trick. Command routines for 'Type', 'Info' and 'Wildcard' appear this month, with more to follow, including file shufflers and a general-purpose viewer for all types of QL graphics file.

Every Qdos enthusiast has individual requirements, so it is valuable to be able to extend the menu to include further operations, such as calls to external programs like editors and archival utilities. This is practical for utilities that accept file parameters via EXEC, like EDT and the GST Assembler, and commands that work on a named file. The 'Type' command uses MORE to display text files.

## Display modes

The Browser can be configured to run in any QL display MODE or character size, including double-height and double-width text. The example screens show it running in a range of CSIZEs.

The two file requesters fit side-by-side on the screen if narrow CSIZE 0,x characters are used, or one above another for Mode8 or wider characters. The default window positions leave 24 pixels free at the left edge of the screen, so they should be readable on most TV screens as well as monitors.

This month's listing is an extension for the file requester routines introduced in issues 8 and 9 of QL World last year, and should be merged with that program. Next issue I shall reveal the remainder of the Browser's own code, and its interface to the requester routines.

I have split the original REQUEST$ function in two, to make it easy to use two windows at once. NEW_VIEW opens the file-name and scroll-bar windows, then reads and displays the directory. SELECTOR allows file operations to be performed once such a view has been prepared.

All the new routines start with WINDOZE (well, it seemed a good idea at the time) which opens the specified number of views. Lines 5160 and 5200 open windows suitable for small Mode4 characters. If you want bigger text, comment out these lines with REMark and replace them with the alternative ones, on the previous lines, removing the "REMark MODE 8:" prefix.

The second parameter is the space to leave clear at the left edge of the screen, and the fourth is the top margin, also in pixel units, followed by the number of lines of text. The examples use the same size for each window, but you can have one tall one and one squat one if your prefer.

PROCedure LOOK_AT (a.k.a. GIMME) puts the path names under each columnar view. Alter the AT commands at the start of each SELect case if you are using two windows in one column. If so, you may like to print the path text in window #1 rather than #0, so it still appears under each view. SELECTOR has only five parameters; it does not need the pixel margins as its windows are ready-made by NEW_VIEW.

The programming has been complicated by the need to ensure that the Browser runs reliably on an any QL interpreter or compiler. It would have been shorter, and simpler, to write a version specifically for Minerva SuperBasic, but that would rule out many users.

Each requester uses more private variables than Sinclair's interpreter can pass safely to a SuperBasic procedure, so some are stored and restored in arrays between one call and the next. There are other possible ways of doing the job, but I chose this approach to minimise surgery in the body of the file requester.

ID% is a square array of four integers, holding the two channel numbers for each view. Increase the first subscript if you want more than two views at a time. PLACE% stores three variables used in each view: LINE%, the current display line, HERE%, the position in the directory arrays, and FREE%, the index of the first un-used array element. SAVE_WINDOW_SPEC stores these values; LOOK_AT sets the global variables from the corresponding array slice.

Two utility functions follow. PADDED$ takes a string and a length and returns the string, packed out with spaces to the length specified. PREFIX takes a string and returns the part up to the first underscore. This is needed because of the way QJump directory entries contain the entire path, less the device name. PREFIX$ gets the drive name from the directory path, to make a complete path for COPY, RENAME and so on.

```
5640 RESTORE 5700
5650 READ com_count%
5660 DIM com$(com_count%,16)
5670 FOR i=1 TO com_count%
5680   READ com$(i)
5690 END FOR i
5700 DATA 13,"ALL","COPY","DELETE","EDIT","GRAPHICS"
5710 DATA "INFO","MOVE","PRINT","QUIT","RENAME"
5720 DATA "SWAP","TEXT","WILDCARD"
5730 END DEFine READ_COMMANDS
5740 :
5750 DEFine PROCedure COMMANDS(this1%)
5760 PAPER #0,0 : CLS #0 : INK #0,7
5770 PRINT #0;PADDED$(path1$,name_Limit%)
5780 AT #0,0,name_Limit%+5 : PRINT #0;PADDED$(path2$,name_Limit%)
5790 PAPER #0,0,2,0 : PRINT #0;"Qdos Browser © SNG 1994"!
5800 PAPER #0,0
5810 FOR i=1 TO com_count%
5820   PRINT #0!!! : INVERSE #0
5830   IF this1%=i THEN PAPER #0,4
5840   PRINT #0!!CHR$(32) & com$(i) & CHR$(32)!
5850   IF this1%=i THEN PAPER #0,7
5860   INVERSE #0
5870 END FOR i
5880 PAPER #0,0
5890 BLOCK #0,480,1,0,19,0
5900 BLOCK #0,480,1,0,29,0
5910 BLOCK #0,480,1,0,39,0
5920 END DEFine COMMANDS
5930 :
5940 DEFine PROCedure DO_COMMAND
5950 LOCal k$(1),j
5960 k$=CHR$(k)
5970 REMark Now try to match K$ with a command
5980 FOR i=1 TO com_count%
5990   IF com$(i,1)==k$ : com_code%=i
6000 END FOR i
6010 IF com_code%<>0
6020   COMMANDS com_code%
6030   IF picking : DO_BATCH
6040   COMMANDS com_code% :REMark Update menu
6050 END IF
6060 END DEFine DO_COMMAND
6070 :

7990 REMark "TEXT" action routine
8000 SAVE_SCREEN
8010 INK #0,4
8020 PRINT #0;"DIY Toolkit MORE file viewer - use keys"
8030 PRINT #0;"↓↑ to move, +ALT for pages, ESC to Exit"
8040 INK #0,7
8050 MORE drive$ & file$ : RESTORE_SCREEN : CURSEN #0 : RETurn
8060 :
8090 REMark WILDCARD cumulative selection
8100 CLS #0
8110 INPUT #0;"Enter wildcard pattern to match:"!wild$\\
8120 IF LEN(wild$)>0
8130   FOR i=1 TO free%-1
8140     IF wild$ INSTR dir$(i)
8150       flag%(i)=flag%(i) || pick_mask%
8160       PRINT #0;dir$(i)
8170     END IF
8180   END FOR i
8190 ELSE
```

## Command Names

The command names are stored in alphabetical order in DATA statements. The initialisation routine READ_COMMANDS copies them into COM$, a string array, to save time later. The first DATA value is the number of commands, followed by their names. They should all start with a different character, as that's how they will be selected from the keyboard, but apart from that the text is up to you.

PROCedure COMMANDS displays the commands in window #0, using INVERSE from DIY Toolkit and some BLOCK commands to set them apart. The most recently selected command THIS1% appears in a contrasting colour. The PRINT exclamation mark separator is liberally used to keep commands from spilling across the ends of lines, but you may need to alter the size of window #0 if you add commands or use big characters.

DO_COMMAND is called from within the file requester. It attempts to match the key-code, K, with the first characters from the command array. If speed is vital you could use INSTR to search a specially-made string holding all the first characters, but in practice this loop seems fast enough; it only runs when the user deigns to press a key. Command codes in COM_CODE% range from one to the maximum set in the DATA. Zero means that no command is selected.

The bulk of the command routines has been held over till next month, but there are a couple of examples to give you a feel for the way they work. Each action is performed by a subroutine chosen with the pidgin-BASIC ON..GO command, which is crude but much faster than SELect or IF tests.

Line 8000 marks the start of the 'Type' command, which gives a full-screen paged display thanks to DIY Toolkit's MORE extension. SAVE_SCREEN and RESTORE_SCREEN buffer the old display in 32K of memory, so that it can instantly re-appear when MORE has finished.

The SCREEN_RAM function detects if Minerva's second screen is in use and makes sure that the currently-displayed screen gets saved and restored. These routines are quite likely to come in handy in other programs. Toolkit 2 extensions ALCHP and RECHP are used here to allocate and release memory; there are many alternatives.

The subroutine at line 8100 performs the "Wildcard" command, which lets you select all the file names that contain a certain string of characters. For instance, _BAS would match all files with the normal SuperBasic suffix. Rather like Toolkit 2 wildcards, this can be over-inclusive, matching cases like TIDY_BASIC_TASK, but it's still easier than picking them all individually.

SHOW_INFO is another general-purpose routine. It extracts the details from a file's directory header, showing its size and detecting sub-directories, linker files, tasks and their dataspaces. The fifth byte of the header is normally zero, but may be set by Toolkit 3's file-protection feature.

The last routine is AWAIT_KEY, which just prints a message in window #0 and waits for a key-press.

```
8200   FOR i=1 TO free%-1
8210     flag%(i)=flag%(i) && type_mask%
8220   END FOR i
8230 END IF
8240 REDRAW
8250 CURSEN #0 : RETurn
8260 :
8270 DEFine PROCedure SAVE_SCREEN
8280 screen_snap=ALCHP(32768)
8290 IF screen_snap>0
8300   MOVE_MEMORY SCREEN_RAM TO screen_snap,32768
8310   CLS : BORDER 2,5 : PAPER #0,0 : CLS #0
8320 END IF
8330 END DEFine SAVE_SCREEN
8340 :
8350 DEFine PROCedure RESTORE_SCREEN
8360 IF screen_snap>0
8370   MOVE_MEMORY screen_snap TO SCREEN_RAM,32768
8380   RECHP screen_snap
8390   screen_snap=0 :REMark Be careful!
8400 ELSE
8410 REMark   Redraw some other way
8420 END IF
8430 END DEFine RESTORE_SCREEN
8440 :
8450 DEFine FuNction SCREEN_RAM
8460 REMark Second screen is controlled by bit 7 of SV.MCSTAT
8470 RETurn 131072 + 32768 * (PEEK(SYS_BASE+52)>127)
8480 END DEFine SCREEN_RAM
8490 :
8500 DEFine PROCedure SHOW_INFO(name$)
8510 LOCal buffa,type,ch%
8520 buffa=ALCHP(64)
8530 IF buffa>0
8540   ch%=NEWCHAN%
8550   OPEN_IN #ch%,name$
8560   GetHEAD #ch%,buffa
8570   CLS #0
8580   PRINT #0\\file$!"is " & PEEK_L(buffa)!"bytes,"!
8590   PRINT #0!!"as of"!DATE$(PEEK_L(buffa+52)) & ";"!
8600   IF PEEK(buffa+4) : PRINT #0!!"protected"!
8610   type=PEEK(buffa+5)
8620   SELect ON type
8630     =0 : PRINT #0!!"data file."
8640     =1 : PRINT #0!!"task, dataspace"!PEEK_L(buffa+6)&"."
8650     =2 : PRINT #0!!"object link file."
8660     =3,255 : PRINT #0!!"sub-directory."
8670     =REMAINDER : PRINT #0!!"unusual type " & type & "!"
8680   END SELect
8690   CLOSE #ch%
8700   RECHP buffa
8710   AWAIT_KEY
8720 END IF
8730 END DEFine SHOW_INFO
8740 :
8750 DEFine PROCedure AWAIT_KEY
8760 PRINT #0\\"Please press a key to continue ..."
8770 k$=INKEY$(#0,-1)
8780 END DEFine AWAIT_KEY
```

## Further Work

The Browser already works under mouse control if cursor-key emulation is enabled, but it would be nicer if it could read the mouse directly. Then you could drag-select names without using the Shift key as the mouse moves, and jump around the file list by grabbing or clicking on the scroll bar.

Such extensions are possible if we use the DIY Toolkit Mouse PTR functions, as we shall see in next month's exciting installment of SuperBasic in Action. In the meantime, please let us know if there are other commandsyou would like to see added to the Browser's menu.

# The NEW USER GUIDE

# Concepts Section

## Section Thirty-Three

**This month the Concepts Guide continues with QDOS.**

### FILES

QL users tend to have a clearly-defined understanding of what constitutes a file, but every device on the QL appears to Qdos as a **file** of some sort. This is something the QL has in common with the Unix operating system and makes for a very powerful, consistent and compact **device** operating system. Devices such as screens and printers are, in effect, infinitely large files. Real files, bound by the restrictions of the medium on which they exist, have a finite length.

All files, **real and virtual,** share some common characteristics. They must each have a unique **name**. They each have a Qdos **channel** associated with them (with the possible exception of the sound chip, which should be a device but which does not have a conventional channel accessible to SuperBasic). Each channel has a memory block allocated to it to hold Qdos-related information.

From now on we will ignore virtual files such as screens. Conventional files fall into three types. There are **data files** that can contain data created at the command line or in programs, data created by applications like Quill, or SuperBasic programs. The inclusion of the latter might appear odd, but SuperBasic code is data input to the 0,0 job running under Qdos, the SuperBasic interpreter. Data files can be created with the PRINT command to a file open for writing or the SAVE command for SuperBasic program lines. The file contents are retrieved into memory using the INPUT and INKEY$ commands. Super Toolkit II users can also read and write using the GET, BGET, PUT and BPUT keywords.

The second class of file is the **executables** that run either simultaneously with others (multi-tasking) or which grab the attention of the system until they are closed (single-tasking). This class of file can only be run using the extended family of EXEC commands. About the only way ordinary SuperBasic users are able to create executable files is through the Turbo compiler.

The third class are repositories of **sections of QL memory** saved with the SBYTES command and retrieved with the LBYTES command. The contents of memory thus saved might be the default fonts, the screen map, Qdos channel tables, a SuperBasic program in its tokenised state, or (most likely) a binary program that can be invoked with the CALL command.

Files can be stored on microdrives, floppy disks or on ram disks, which are areas of working memory formatted to act exactly like microdrives but unable to store information without constant power. Storage media are formatted into blocks or sectors of a pre-determined size. Large files will occupy several sectors. One disadvantage of this arrangement is that a whole number of sectors must be allocated to a file: if there is space left over in a file's final sector it is unusable. Storing a few large files is more space-efficient than storing several small files because the number of partly-used sectors is so much lower.

Files are given a header of 64 bytes, broken down as follows:

**00 - 03An integer representing the file length**
**04 - 04The file access key**
**05 - 05The file type byte**
**06 - 13Eight bytes of file information**
**14 - 15The length of the filename (up to 36 bytes)**
**16 - 51Reserved to hold the filename**
**52 - 55(disks only) Date stamp**
**52 - 55(microdrives) not used**
**56 - 63Reserved**

The **file access byte** is normally set to zero. The **file type byte** is set to one for executable files and is zero for both of the other file types. For executable files the first four bytes of the file information block form a long word integer representing the size of dataspace to be allocated to the task as soon as it is launched. Super Toolkit II and Minerva implement date stamping when creating and updating files on microdrives, but neither can alter the inherent dating behaviour of floppy disk controllers.

Microdrives and floppy disks do not just contain conventional files. The first sector of a microdrive, for instance, is reserved to hold a **map** showing which segment of which file is stored in each of the other 512-byte sectors. Microdrives also need directory files, normally invisible to the user, containing copies of the file headers for each of the files on the medium. Qdos also avoids writing to bad sectors by assigning them to a non-existent and unreachable file.

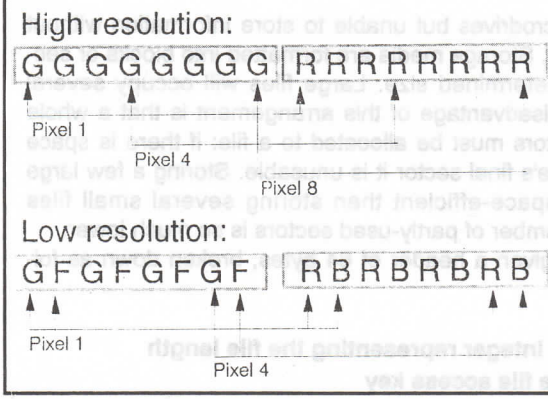
When microdrive tapes are formatted the sectors are created and the **sector map** and directory entry are written. The reported total sector count is usually around 220 sectors. If you are lucky, the available sectors will be exactly two less than the maximum: one sector is taken up by the microdrive map and the other by the directory. The length of microdrive tapes must be kept within a maximum size of 255 sectors otherwise the mapping conventions would break down. No matter how carefully you treat your microdrives, one sector is always going to be bad

- the one where the tape is spliced - and it is ignored in the sector count.

**Floppy disk formats** are more complicated than microdrives to take account of their several tracks and two sides. Differences include the following: areas of disks are allocated in clumps of three sectors at a time, the file headers still exist but are more or less redundant in favour of the directory map, and after formatting a perfect 1440 sectors is normally achieved, 6 of which are immediately allocated to the disk map and the file directory.

Users may have noticed something slightly spooky when running small programs on QLs with large memories. After spending an age finding the first program on a microdrive, it suddenly seems that only the briefest examination of a drive is necessary to load the next program or piece of data. This is because Qdos does its level best only ever to read microdrive and disk sectors once. It copies images of all sectors it happens to pass across into any ram that is otherwise unoccupied. If the user requests access to a file stored on sectors paged into memory, Qdos goes to the memory image rather than the original source. As memory access is considerably faster than disk or microdrive access the result is a significant acceleration in loading speeds. The brief twitch of the drive light is Qdos checking that you have not sneakily swapped one microdrive for another while it was not looking. Of course, as you fill up ram with multi-tasking programs or large data structures there is less room for Qdos to squirrel away copies of the microdrive sectors and so it must resort to the slow task of reading directly from the medium.

## Screen map layout - first 2 bytes




## IDENTIFIERS

**Loop identifiers** and **variables** can look very similar, but there are significant differences. Every REPeat loop needs to be given a name, using the same rules as those governing numeric variable names. Other languages make do without loop names, but they demand an explicit exit from each level of a nested series of loops. In SuperBasic, if you are in the deepest level of a set of nested loops you can escape with a single bound using just one EXIT command, as long as it mentions the outermost loop's name.

FOR...NEXT loop identifiers have a specific numeric floating point value that can be used within the loop. REPeat loops have no such luxury, so a loop counter must be created and maintained explicitly. The variable used in a SELect structure has this same restriction of being a floating point number. Minerva users and creators of compiled **Turbo** code can ignore these restrictions. In the case of Turbo, it is well worth the effort of fooling SuperBasic into using integer variables for loops: the acceleration is dramatic.

## MEMORY MAP

Every location in the QL's chip-based memory has an address, beginning at 0. The first 48K is reserved for **system rom**, the internal layout of which changed significantly to put language-specific data such as fonts and error messages in the higher addresses. The next 16K is allocated to **rom boards** plugged into the back of the QL. There follows an area of 8K reserved for input and output. The screen map, 32K in size, follows that. The rest of addressable ram is then apportioned between the **user, Qdos and microdrive slave blocks.** Things placed within this area have a habit of being shunted around as Qdos re-organises the memory to meet new demands.

**Qdos grabs** a significant part of the QL's available ram, most of it located immediately after the screen map beginning at location 163840. It begins with a tableaux des tableaux that includes pointers to other tables that may in turn point to yet other tables. **Tables** might contain device channel information, variable names and values, keywords, and so on. SuperBasic programs are stored near the top of available ram and are held in tokenised form. Part of the remainder forms a **"heap"** from which memory may be allocated. Unused memory is sectioned into blocks to mirror microdrive storage sectors (see the earlier entry on Files for more details).

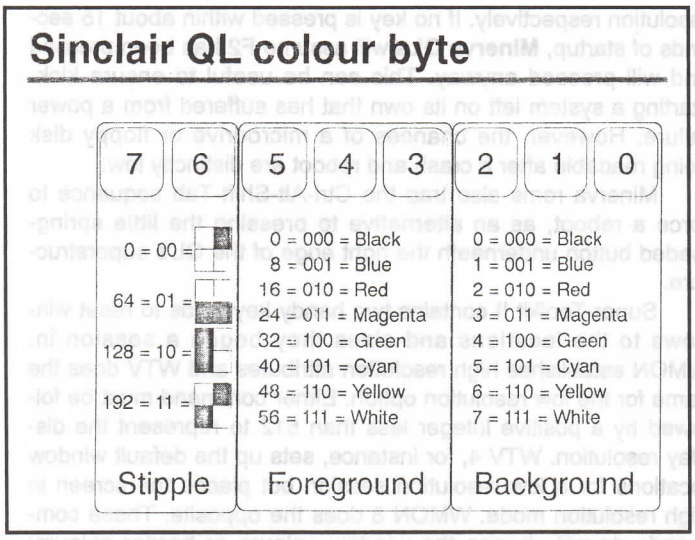
It is not possible to give a comprehensive listing of Qdos memory usage within the format of this guide. Nevertheless, some of the most useful addresses are listed in the first table.

## NETWORK

As with so much else connected with early Qdos, networks did not really work until **Super Toolkit II** arrived. Many QLs before those with a D14 batch number were built and sold without working network ports. Since then, there have been several interesting success stories with connecting several QLs - even with some of the notoriously unco-operative pre-D14 production batches. The physical layer of a network is rudimentary in the extreme: bell wire is connected to the network ports behind the microdrive housing of all of the QLs partaking in the exercise to form a circuit. This is usually the easy bit.

## Pure colour combinations

| Colour | Green | Red | Blue | Value |
|---|---|---|---|---|
| Black | Off | Off | Off | 000 = 0 |
| Blue | Off | Off | On | 001 = 1 |
| Red | Off | On | Off | 010 = 2 |
| Magenta | Off | On | On | 011 = 3 |
| Green | On | Off | Off | 100 = 4 |
| Cyan | On | Off | On | 101 = 5 |
| Yellow | On | On | Off | 110 = 6 |
| White | On | On | On | 111 = 7 |




Data is transmitted across the network in blocks reminiscent of the way files are blocked onto sectors of a floppy disk or microdrive. Each block has a **header** identifying where it has come from, where it is going, its size and whether or not it is the last block in the set. Sending **network blocks** is exactly like sending Morse code with a torch down a tunnel: everyone looking into the tunnel can see flashing lights, but if two signallers try it at once the result is confusion. The network is managed by a protocol - a set of rules that determines who can send what and when. The stations transmit "scouts" to see if the network is free and there is someone listening. If the scout is successful, blocks get transmitted. Even though the blocks may be destined for one particular QL, every other QL on the network detects the signal, but ignores it.

At the operating system level, QLs are joined to the network by channels. The following commands represent some the options available:

**OPEN #6, neti_3_12: REMark input channel, 3 is this QL's identifier, 12K buffer**
**OPEN #7, neto_7: REMark output channel to QL station 7**
**OPEN #8, neto_0: REMark general broadcast to everyone else**

With Super Toolkit II installed, input channels can have a buffer size allocated to them, otherwise the network will grab almost all the free memory available. If you try networking without Super Toolkit II (surely a recipe for frustration) this refinement is not available. Once the channels are open, files are transmitted with the COPY family of commands.

Each computer (or **"station"**) on the network has to have a unique identifier so that it can recognise data blocks intended for itself and ignore blocks destined for other computers. Where there are only two QLs in the net this is not an important factor, provided they only open **one input channel and one output channel each** to the network, because they can both use the default station number of 1. When three or more are gathered together, each QL should be assigned a station number with a command such as OPEN #6, neti_4. Station numbers must be integers within the range 1 - 63.

A much more valuable mode of network operation can be obtained by invoking Super Toolkit's **file server task**, FSERVE. This multi-tasking utility sits in the background connecting remote QLs to up to ten resources (windows, printers, disk drives, etc) on the file server. Filer servers must have a network identifier in the range 1 to 8, and the lower the better. The file server is implemented by issuing the FSERVE command. To all the other computers on the network, the file server appears as a two character prefix placed before any normal channel information, such as:

**OPEN #6, n3_con_200x40a0x0: REMark open a window on the file server**
**INK#6, 4: PAPER#6, 0: INPUT #6, "Can you hear me, Mother?", reply$**
**OPEN#7, n3_ser1: PRINT #7, "This is output to a remote printer"**

Even better, remote devices can be given what appear to be local names using the NFS_USE command. This works exactly like the more familiar FLP_USE by assigning a virtual drive name to a remote actual drive. FLP1_ on a file server can become MDV1_ by issuing the command:

**NFS_USE mdv, n3_flp1_, n3_flp2_**

In practice, FSERVE is a **device server** of great flexibility and real benefit. About the only weakness (and this applies to all networks no matter what the computer) is the possibility that a remote station is turned off with files open on a remote server. The potential for damage is reduced by the user of the file server explicitly cancelling the SERVER job before turning off the computer using the RJOB SERVER command.

## QDOS

The QL's operating system contains device drivers, permits multi-tasking and it was designed to be extendible right from the outset. It was demonstrably far ahead of the version of Microsoft DOS current at the QL's launch and has several advantages over even the latest version of MS-DOS. Given Sinclair Research's initial distaste for such conventional things as floppy disk drives, it seems unlikely that Qdos was intended to stand for QL Disk Operating System. Some prefer QL Device Operating System, but it is not clear that this was thought of when the QL was launched: early publicity material studiously avoided giving any decode of the initials. Perhaps it stood for nothing very much but was designed to recall to people's minds MS-DOS and the general DOS concept.

An unusual but very useful feature of Qdos, almost always overlooked, is its **integration** with SuperBasic, saving on duplicating file handling facilities and encouraging non-programmers to type in at least a few commands at the command prompt.

## SCREEN COLOURS

Colour is represented within the QL at two levels: firstly, all foreground/background/stipple colour combinations can be represented by a single byte and, secondly, coloured pixels are represented in a raster scan scheme within the screen map. This topic deals with both representations in turn. Note that foreground and background colours are not the same as INK and PAPER, but refer to the combinations of colours that make up stippled shades.

All possible colours on any monitor or TV screen are created by mixing light from combinations of three potential sources, one red, one green and the other blue. In natural-colour systems, such as televisions, the individual colour sources can be fired at different strengths, but for the QL the options for each source are either on or off, leading to just eight colours.

The **table of colour combinations** shows that the eight colours can be represented using three bits. Within any colour byte the last three bits represent the background colour and the next three bits represent the foreground colour. This occupies

six of the eight bits in a byte, leaving the first two bytes free to represent something else. On the QL, the first two bits of a colour byte identify the stipple used to mix the foreground and background colours.

SuperBasic allows **colour statements** to have up to three parameters, representing background, foreground and stipple. PAPER 5, for instance, produces solid cyan and INK 4, 2 produces an even stipple of red and green. There are some instances where instead of three parameters it can be useful to declare the full colour combination with only one value. This can be done by referring to the **Colour Byte table** and adding the values together for the required stipple, foreground and background. PAPER 167 will produce stripes of green and white, for example.

## Some useful QDOS addresses

| | | |
|---|---|---|
| 163856 | Long | Base of SuperBasic area |
| 163886 | Word | Random number |
| 163890 | Byte | Holds 1 if in TV mode |
| 163891 | Byte | Holds 0 if screen active, 1 if frozen |
| 163895 | Byte | Network number |
| 163900 | Long | Pointer to polled task list |
| 163904 | Long | Pointer to scheduler tasks list |
| 163908 | Long | Pointer to device drivers list |
| 163912 | Long | Pointer to directory devices list |
| 163916 | Long | Pointer to start of keyboard queue |
| 163960 | Long | Pointer to start of channel table |
| 163976 | Word | Caps lock: 255 = on, 0 = off |
| 163980 | Word | Key auto-repeat delay |
| 163982 | Word | Key auto-repeat rate |
| 164078 | Byte | Currently active microdrive |

The way that screen colours are represented in the QL's memory map makes it difficult to control colour directly in the memory map. In low resolution mode, each set of four pixels share 16 bits-worth of data in the memory map. To reach high definition, eight pixels have to share the same amount of data, which reduces the range of available colours by half but doubles the number of pixels that can be addressed. The virtual screen that all of the window-related commands recognise is a constant 512 pixels by 256 pixels, but in low resolution mode the real screen is only 256 pixels wide by 256 pixels deep. Figure four shows the relationship between bits in the screen map and coloured pixels on the screen. Note how Sinclair made use of the "spare" bit in low resolution mode to signify flashing.

There have been moves to add another screen resolution to the QL for some time and it may soon become reality for Miracle Super Gold Card owners. This will be a godsend to users of programs with a lot of detail like Professional Publisher. Keep reading QL World for details.

## START UP

When a QL with a Sinclair rom starts up or is reset the screen offers two choices: press F1 to begin a high-resolution session or F2 for a low-resolution session. Although Sinclair recommended using F2 only on televisions, either option can be selected whether you are using a monitor or a television, although windows sizes and locations may need adjusting if they are not to bleed off the picture on a TV set.

When a QL has its rom replaced by a Minerva, the start-up sequence offers slightly different choices. F1 and F2 work as before. F3 and F4 start up a second screen using high and low resolution respectively. If no key is pressed within about 15 seconds of startup, **Minerva QLs** will assume F2 has been pressed and will proceed anyway. This can be useful to ensure kick-starting a system left on its own that has suffered from a power failure. However, the chances of a microdrive or floppy disk being readable after a crash and reboot are distinctly low.

Minerva roms also trap the Ctrl-Alt-Shift-Tab sequence to force a reboot, as an alternative to pressing the little spring-loaded button underneath the right edge of the QL's superstructure.

Super Toolkit II contains two handy keywords to reset windows to the locations and sizes they began a session in. WMON establishes high resolution attributes and WTV does the same for the low resolution option. Either command must be followed by a positive integer less than 512 to represent the display resolution. WTV 4, for instance, sets up the default window locations for a low resolution session but places the screen in high resolution mode. WMON 8 does the opposite. These commands do not change the window colours or border colours: these have to be adjusted using other SuperBasic commands.

## WINDOWS

A window is an area of the screen that operates independently. Unfortunately, when windows overlap on the screen there is no sense of one window being "on top" of another window. In Qdos, whichever window is written to last will **overwrite** whatever the other screen had previously displayed. Colour, scrolling, character sizes and so on are controlled separately for each window. Qdos keeps track of these attributes using a special channel table for each window.

There are in fact two distinct types of window recognised by Qdos, those that accept input and those that do not. Windows in which the INPUT command does something are called **consoles**; windows in which the INPUT command is ignored are called **screens.** A window is defined as being either a console or a screen when its channel is opened. All of the default windows opened when the QL is booted are consoles.

Screen referencing uses one of **four co-ordinate systems:** screen, character, block and graphics. The **screen** co-ordinate system is used to define the location of windows and has its origin at the top left corner of the screen. The origin for both the **character** and the **block** co-ordinate system is at the top left corner of a window, while that for graphics is at the bottom left corner of a window unless it is moved with the SCALE command. Block co-ordinates directly correspond to pixels and are used by the BLOCK command. Character co-ordinates are related to character sizes set by the CSIZE command. **Graphics** co-ordinates reflect the current setting of the SCALE command and so unlike the other co-ordinate systems is not based on pixels.

Some screen-related commands, such as PAN, SCROLL and CLS, recognise window parts using the following codes:

**0 The whole screen**
**1 Above the cursor line (but excluding it)**
**2 Below the cursor line (but excluding it)**
**3 The cursor line**
**4 The cursor line to the right of the cursor, including the cursor**

# The WORLD at your fingertips

**Hilary Snaden has converted a PC map-generator for the QL**

Compared to other machines the QL has been relatively poorly endowed with public domain software, and it was in search of transportable Basic programs that I looked at the large catalogues of MS-DOS PD. What caught my eye instead was a collection of data called **The World Digitised**, described as "the co-ordinates for 100,000 points on the Earth's surface including the latitude and longitude of the geographical borders and coastlines."

This looked like an excellent way of producing good quality maps on-screen without poring over atlases, trying to work out the co-ordinates, and then typing them in. This article is intended to explain some of the detective work behind making this particular set of data accessible to the QL, and to give some ideas on how to use it.

## MS-DOS disks

The first problem on receipt of the three disks was, of course, that they were in MS-DOS format. Perhaps the best-known utility for transferring files between different disk formats is **DiscOver.** However, I used **IBMDisk**, an easy-to-use public domain program by Jonathan Hudson which copied the files to Qdos disks with no trouble.

It should be mentioned that while, when transferring text files (such as documents or Basic programs) from MS-DOS disks the transfer utility can be configured to remove the CHR$(13) - which MS-DOS adds to the end of each line and which Qdos does not need - archived files must be transferred unchanged otherwise no unarchiving utility will make much sense of them!

## Unarchiving

The three disks contained a total of nine archived files in addition to some text files of advertisements for the library which supplied the discs, all of which could be safely ignored. Of the nine archived files, two contained documentation and sample programs (for MS-DOS machines), while the remaining seven contained the data, one file for each continent

The archiving of the essential files was the next problem to surmount. Files are commonly archived to save disk space; in this case the method was the fairly well-known ARC.

At the time of writing there are two utilities available which will un-ARC files, both in the public domain. Unfortunately one of these refused to recognise the ARCed files, but Jan Bredenbeek's **UNARC** program worked with no problems, and the seven archived files which had occupied 706K (and so conveniently fitted - just - on a standard 720K Qdos disc) had become thirty files occupying 938K.

Of these files the only ones which needed to be kept were the documentation (the example Basic handling programs were less useful than they might have been, and some were not text files) and, of course, the data itself.

Having extracted the data files from their archives, there is another problem; the geographical co-ordinates were not numbers in a text form but in a binary form.

These binary files, all with the extender MPS contained data in IEEE short real format, a "standard" way of packing numbers into four bytes about which, unfortunately, Qdos knows nothing.

Source code in C for a program to convert these binary files into numbers as text (and therefore readable by SuperBasic) was among the documentation, and after perusal of the code and a sample text data file which had helpfully been supplied, and some experimentation, an equivalent SuperBasic program was devised.

## Adaptations Needed

**Listing one** shows the main part of this, a function which takes a four-byte string and returns in floating-point format the number that it represents. Note that owners of non-Minerva QLs would need to adapt these programs, as lines such as FOR i%=1 TO 4 and SELect ON I% will not work at all on non-Minerva roms.

In short real format, zero is represented uniquely by four null bytes, and the function checks for this separately.

Each co-ordinate therefore consisted of eight bytes (four for each ordinate) separated by line feeds. In addition blocks of points were separated by a CHR$(1), and some lines contained in addition text strings in the manner of REMs, explaining which areas of the world were covered by that file or block of data.

Hence the conversion program has to input a string from the MPS file and check its length. If it is longer than nine bytes it is a line containing a text message; the first eight bytes can be converted to two co-ordinates in floating point format, and written out to the output file with a space separating the ordinates and with the original text message appended. The output files containing the data in a text format were suffixed _MP0 as suggested by the documentation.

If the input string is nine bytes long, the ninth byte is a CHR$(1), a character used to separate blocks of connected points; the first eight bytes are then converted and written to the output file in the normal way and then followed by a blank line (the block separator in the MP0 files). CHR$(1) may occur at the end of a longer line containing text, and this also has to be checked for.

## Extra CHR$(10)

If, however, the string is less than eight bytes long it means that the eight bytes

Listing one: Converting IEEE short real format to floating point

```
520 DEFine FuNction real(t$)
530 LOCal t%(4),s,m,e,n,n%
540 IF t$=FILL$(CHR$(0),4) : RETurn 0
550 FOR n%=1 TO 4 : t%(n%)=CODE(t$(n%))
560 s=1 : IF t%(4)>127 : t%(4)=t%
(4)-128 : s=-1
570 m=((t%(3)MOD 128)*65536+t%(2)*
256+t%(1))/2^23
580 e=t%(4)*2+(t%(3)DIV 128)-127 : e=2^e
590 n=m*e+e
600 n=INT(1000*n+.5)/1000
610 IF s=-1 : n=-n
620 RETurn n
630 END DEFine real
640 :
```

Listing two: Converting MPS files to MP0$$

```
100 PRINT 'MPS to MP0 file conversion'
110 INPUT 'Input file:'!infile$
120 INPUT 'Output file:'!outfile$
130 OPEN_IN#3,infile$
140 OPEN_NEW#4,outfile$
150 decode
160 CLOSE#3 : CLOSE#4
170 :
180 DEFine PROCedure decode
190 LOCal t$,x$,y$,l%,n%,i%
200 REPeat outer
210   IF EOF(#3) : EXIT outer
220   INPUT#3,t$ : l%=LEN(t$)
230   SELect ON l%
240   =0 TO 7
250     REPeat inner
260       INPUT#3,v$
270       t$=t$&CHR$(10)&v$
280       l%=LEN(t$)
290       IF l%>7 : EXIT inner
300     END REPeat inner
310   END SELect
320   IF l%>9
330     z$=t$(9 TO l%) : t$=t$(1 TO 8)
340   END IF
350   x$=t$(1 TO 4) : x$=real(x$)
360   y$=t$(5 TO 8) : y$=real(y$)
370   SELect ON l%
380   =8 : PRINT#4;x$!y$
390   =9 : PRINT#4;x$!y$ : PRINT#4
400   =REMAINDER
410     z%=LEN(z$)
420     IF z$(z%)=CHR$(1)
430       PRINT#4;x$!y$;z$(1 TO z%-1)
440       PRINT#4
450     ELSE
460       PRINT#4;x$!y$;z$
470     END IF
480   END SELect
490 END REPeat outer
500 END DEFine decode
510 :
```

which should be there have become split over more than one line because one of the binary data bytes happens to be a line feed. In this case the CHR$(10) needs to be appended at the end of the string and the remaining bytes which are part of that co-ordinate's data read in using a REPeat loop until the string is eight or more bytes in length.

It sounds complicated; **listing two** shows how it was done. The conversion process was horribly slow, as SuperBasic was being used for a vast amount of data to be converted, but at least it had to be done only once for each of the data files.

One should beware of using Supercharge or Turbo to compile programs like this as their enhanced numerical accuracy can in fact introduce rounding errors which occupy disk space with redundant digits, and you will need a further function such as Toolkit II's FDEC$ to restore the three decimal place accuracy of the original data.

The end result of all this is that we have thirty Qdos-readable data files, now occupying 1537K. We can now start making use of the information they contain.

## Organising Disks

The most difficult data problem was a logistical one, caused by the sheer size of the data files. If you have a hard disk, or ED drives and disks, all the files will fit on a single ED disk, but lesser mortals will need to give some thought to choosing files and organising them to avoid having constantly to swap between data disks.

For example, if you are interested in the outlines of continents and islands, you could gather together the files numbered 0 and 1 (files numbered 2 and 3 cover lakes and national borders); to cover a smaller area but including national boundaries and lakes, all the files for that area could be put together, for example e0 to e3 for Europe, af0 to af3 for Africa, and so on.

The data files now all looked something like **Example one**. After all the work just described we are at last on easier ground as the data is in a recognisable and accessible format. Drawing detailed maps on the QL screen is simply a matter of setting up a screen window with appropriate size, scale and origin, opening a channel to the required data file, reading in the data and plotting it using POINT.

In reading the co-ordinate data the inputted strings need to be sliced to extract each ordinate, with the blank lines that separate blocks of connected points being ignored.

## Map Drawing

**Listing three** draws a map of Europe in this way. The dimensions and scale of the screen window are somewhat arbitrary, the result of a fair amount of experimentation to produce a reasonably authentic-looking map. Due to the time taken for the plotting programs to complete their work (the world map took about 45 minutes to draw with a JS rom) the completed screens were saved with SBYTES filename,131072,23808. Other programs can then use the maps without having to wait so long by loading them back into screen memory with LBYTES filename,131072 and using the same screen dimensions as were used in drawing them.

It is important not to use too much of the data as plotting too many points on a small-scale map will obscure rather than show detail.

The map of the world was drawn using every seventh point, except for the coastline of Antarctica, for which every other point was plotted. Even at larger scales, as in the map of Europe, the points are only just beginning to become obtrusive.

Only when dealing with relatively insignificant areas of the world such as Britain will it be necessary to join the points using LINE. The map of the UK was plotted in this way, and here two nested loops were needed in reading the data to avoid drawing lines between points which should not be connected. **Listing four** is this part of the program which produced the map of the UK.

The original documentation describes how to produce

Listing three: Simple plotting program

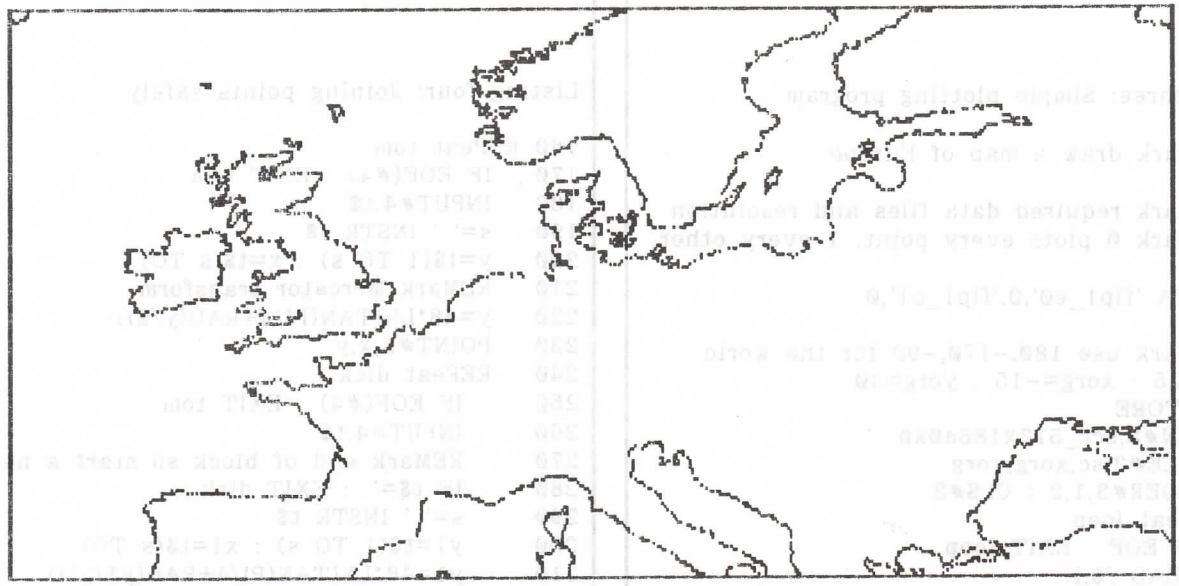
```
100 REMark draw a map of Europe
110 :
120 REMark required data files and resolution
130 REMark Ø plots every point, 1 every other,
140 :
150 DATA 'flp1_eØ',Ø,'flp1_e1',Ø
160 :
170 REMark use 18Ø,-17Ø,-9Ø for the world
180 sc=25 : xorg=-15 : yorg=4Ø
190 RESTORE
200 OPEN#3,scr_512x186aØxØ
210 SCALE#3,sc,xorg,yorg
220 BORDER#3,1,2 : CLS#3
230 REPeat loop
240   IF EOF : EXIT loop
250   READ f$,i
260   f$=f$&'_mpØ'
270   plotfile f$,i
280 END REPeat loop
290 STOP
300 :
310 DEFine PROCedure plotfile(file$,n)
320 LOCal tom,i,p
330 OPEN_IN#4,file$
340 REPeat tom
350   IF EOF(#4) : EXIT tom
360   INPUT#4,t$
370   IF t$='' : NEXT tom
380   s=' ' INSTR t$
390   y=t$(1 TO s) : x=t$(s TO)
400   IF x>19Ø : x=x-36Ø
410   IF x<-17Ø : x=x+36Ø
420   POINT#3,x,y
430   IF n
440     FOR i%=1 TO n
450       IF EOF(#4) : EXIT tom
460       INPUT#4,t$
470       IF t$='' : NEXT tom
480     END FOR i
490   END IF
500 END REPeat tom
510 CLOSE#4
520 END DEFine plotfile
530 :
```

Listing four: Joining points safely

```
160 REPeat tom
170   IF EOF(#4) : EXIT tom
180   INPUT#4,t$
190   s=' ' INSTR t$
200   y=t$(1 TO s) : x=t$(s TO)
210   REMark Mercator transform
220   y=48*LN(TAN(PI/4+RAD(y/2)))
230   POINT#3,x,y
240   REPeat dick
250     IF EOF(#4) : EXIT tom
260     INPUT#4,t$
270     REMark end of block so start a new line
280     IF t$='' : EXIT dick
290     s=' ' INSTR t$
300     y1=t$(1 TO s) : x1=t$(s TO)
310     y1=48*LN(TAN(PI/4+RAD(y1/2)))
320     LINE#3,x,y TO x1,y1
330     x=x1 : y=y1
340   END REPeat dick
350 END REPeat tom
360 STOP
370 :
```

Example one: Part of an MPØ file

```
80.438 97.373 The World Digitized
80.424 97.352 Islands of Europe
80.407 97.336 Copyright 1986 John B. Allison
80.390 97.314
80.370 97.260
80.353 97.195
80.305 97.098
80.202 96.748
80.158 96.559
80.138 96.430
```



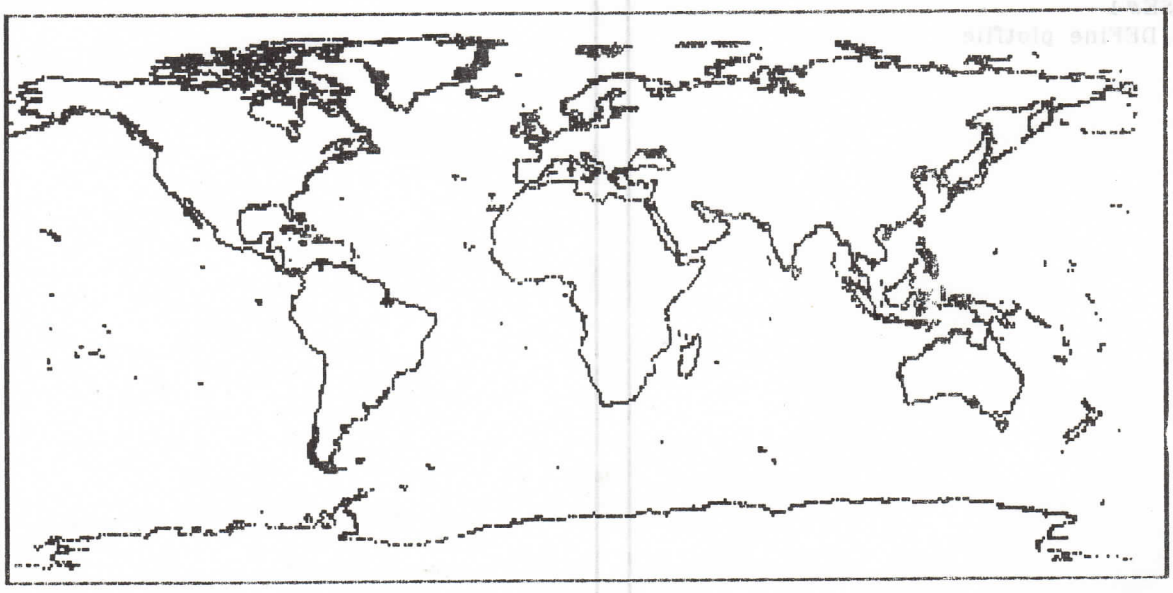
Mercator projections; that was not done with the maps of the world and Europe as the rather messy trigonometry required would have further slowed the map drawing, and needed the same transformation to be written into a program which I intended to use the screens. The map of the UK shows how using a Mercator projection produces a shape more like the one we are used to seeing.

The documentation on the disks gives some practical ideas on using the data, and there are doubtless a good many more which SuperBasic hackers can devise for themselves.

## Files For QL?

If there is sufficient interest it may be possible to make the converted data files described here, together with the original documentation, available through a QL public domain library. Otherwise, **The World Digitised** is available as volume number 1094 from **The Public Domain Software Library, Winscombe House, Beacon Road, Crowborough, Sussex, TN6 1UL. The cost was £13.95 for three 5.25-in discs, £15.45 for 3.5-in, but check prices before ordering.**

**From top: Middle Europe; British Isles; The World.**



# DISKCOPY

## Neil Gordon addresses the question of backups.

ONE of the most important things with computer media, whether microdrives or disks, is to keep backups of everything. This is essential since data can become corrupted due to media failure, or destroyed by an accident such as formatting the wrong disk! Making backups is easier said than done, especially if you only have one disk drive and the disk capacity is greater than the computer's memory. To deal with this problem I wrote this program.

### Check the System

Please note that the following is based on my system, which is a QL with 512K Expanderam memory expansion, and a Cumana disk interface with one 3.5 inch, double density, double sided disk drive attached, however it should be applicable for most disk users since most disk interface toolkits supply the same basic commands. The commands used are the "GET" and "PUT" commands (lines 430 and 610), which can copy sectors from a disk, and copy sectors onto a disk. The expression following each of these supplies the required location on the disk, and these may need altering for other systems - check your manual for details.

The program works by ignoring the file structure on the disk, and simply copying the sectors themselves. The sectors are what contain the data on the disk, a particular file consisting of several sectors.

On the most popular disk systems for the QL, each sector contains 512 Kbytes - one kilobyte containing 1024 bytes - so a disk containing 1440 sectors has a capacity of 720K. To copy the data on a disk, an array (called block$ in the program) is set up. Now clearly, even with the 640K of an expanded QL, there is not sufficient memory to store the whole disk in memory, so the program copies the disk in blocks. To reduce the amount of disk swapping, I have set up the program to copy it in blocks of 360K, but if you have different memory available, the constant ramperswap (line 140) can be set accordingly. Clearly, with the 360K blocks it will require two disk swaps for a 720K disk to be read, different memory or disk capacities will affect the number of disk swaps required, so, for example, copying a 360K disk with

```
100 REMark Neil Gordon. 1991 - Program to copy discs by sectors.
110 REMark Set System specific variables
120 source$='flp1_*D2D'
130 destination$='flp1_*D2D'
140 ramperswap=360:REMark ram in K used a disc swap,<max ram.
150 sectorlen=512:REMark bytes/sector
160 blocklen=INT(ramperswap*1024/sectorlen)
170 nosides=2:REMark number of sides/disc
180 sectorsatrack=9
190 tracksaside=80
200 totalsectors=nosides*tracksaside*sectorsatrack
210 DIM block$(blocklen,sectorlen)
220 presentsector=0:putsector=0
230 WINDOW 400,200,15,15:CLS
240 PRINT 'Disc Copier'\'Present settings:'
250 PRINT totalsectors*sectorlen/1024;'K disc.'
260 PRINT tracksaside;' tracks, ';sectorsatrack;
270 PRINT ' sectors per track, on a ';nosides;' sided disc.'
280 PRINT  'Copied using ';ramperswap;' Kbytes a disc swap'
290 AT 7,1:PRINT 'Side:'\'Track:'\'Sector:'
300 REPeat loop
310  AT 12,0:PRINT 'Insert disc to be COPIED & press a key'
320  BEEP 10000,10:PAUSE:AT 11,0:CLS 2:PRINT \'Loading Data'
330  OPEN_IN #4,source$:startsector=presentsector
340  FOR b=1 TO blocklen
350   presentsector=presentsector+1:REMark record disc position
360   side=(presentsector-1) DIV (tracksaside*sectorsatrack)
370   AT 7,6:PRINT side
380   track=((presentsector-1) DIV sectorsatrack) MOD tracksaside
390   AT 8,7:PRINT track;" "
400   sector=(presentsector MOD sectorsatrack)
410   IF sector=0 THEN sector=sectorsatrack
420   AT 9,8:PRINT sector
430  REMark system specific function to read sector from disc
440   GET #4\sector+side*256+track*65536,i$
450   block$(presentsector MOD blocklen)=i$
460   IF presentsector=totalsectors THEN EXIT b
470  END FOR b
480  CLOSE #4:AT 12,0:PRINT 'Insert "BLANK" disc and ';
490  PRINT 'Press a key':BEEP 10000,4:PAUSE:AT 11,0:CLS 2
500  PRINT \'Saving Data':OPEN_NEW #4,destination$
510  FOR putsector=startsector+1 TO presentsector
520   side=(putsector-1) DIV (tracksaside*sectorsatrack)
530   AT 7,6:PRINT side
540   track=((putsector-1) DIV sectorsatrack) MOD tracksaside
550   AT 8,7:PRINT track;" "
560   sector=(putsector MOD sectorsatrack)
570   IF sector=0 THEN sector=sectorsatrack
580   AT 9,8:PRINT sector
590   i$=block$(putsector MOD blocklen)
600  REMark system specific function to put sector on disc
610   PUT #4\sector+side*256+track*65536,i$
620  END FOR putsector
630  CLOSE #4
640  IF presentsector=totalsectors THEN EXIT loop
650 END REPeat loop
660 AT 11,0:CLS 2:PRINT 'Disc copied successfully.':BEEP 10000,0
```

78K per swap will require 360/78=4.6, that is, 5 swaps.

## Swapping Disks

The "GET" command is used to copy the necessary sectors into the array. Once the first block has been loaded, the disk being copied is removed, and the blank disk can be inserted into the drive. The data in the array is then copied to that disk using the "PUT" command.

Once this first block has been copied, the source disk is again placed in the drive. The above process is then repeated until the remaining disk data is copied. The end result is an exact copy of the original disk.

These commands require a pointer to locate the disk read/write head, so the program variable presentsector keeps a record of the position on the disk, this value being converted to the required track, sector, side information by utilising the DIV and MOD functions.

Directly copying sectors simplifies copying, since there is no need to ensure that file sizes are less than the available memory, and that all files have indeed been copied. The downside of this technique is that some time will be spent copying 'empty' disk sectors, but at least the program will take the same amount of time for every disk copied.

The program contains constants for the data layout on the disk and the number of sides. It should not be too taxing to customise it to your system. Remember - it is better to have more copies of your work than you need than to discover you have lost your only copy!

Please note that copyright must be observed, so check program documentation before copying!

# Hugh In Newport

The first Newport/ Miracle Show sponsored by ILQR is now part of QL history. The second Miracle in Newport will join QL history on 14th May this year. This was how it went in 1993.

On Friday June 4th at the Carlton Motel in Newport we were busy getting to know each other and general gabbing. Many visitors from all over the States were staying there, some at motels nearby.

The UK bunch, Stuart Honeyball, Tony Firshman and Bill Richardson were busy just talking, and there was a fair amount of swapping wares and information.

## Many Meetings

I was able to meet many members of our club, the Toronto Timex-Sinclar Users Club, show them a program I had just completed called Quandrex, which is an index-cross reference thing for the Quanta magazine.

On Saturday I had a table for our club at the Salvation Army Building where the Show was held, meeting many of our members and many who were not. Now I can say that many more know who, what and where we are!

I had my trusty QL with me, and all the stuff that goes with it, with a few demo programs. But shortly into the demo, my QL decided it was a bit shy in that horde, and that it did not like the 1 key. So I replaced the membrane - but my QL was still not up to appearing in public, and decided to stop accepting power at the input socket. Woe is me - a load of demos to run and my trusty QL goes bashful on me. However I was able to borrow a spare unit from Bill Cable. But when I started to install my Gold Card in Bill's machine, I bent the pins in the port. Fortunately I had a pair of long nosed dodads to straightenthe pins.

My demos were mainly run using a program called Vision Mixer (DJC), with many of the Eye-Q (Digital Precision) pictures, plus some I made with PictureMaster (DJC again).Vision Mixer allows you to change screens in a nice easy automatic manner, and the screens change in many different ways, just like TV.

While I was getting sorted out, other parties were going full steam ahead with their own demos and selling. There did not appear to be a lack of the green stuff and many were in a buying mood. (*Note for non-travellers: all American currency notes are green, rather like the old UK £1 notes before the present "art printing" commenced.*)

Stuart Honeyball was there, pushing the QXL card, and doing good business by the number of bodies clustering around his table. The QXL turns the PC into a QL-compatible (now why didn't IBM think of that?). I don't have a PC and I don't imagine I'll be getting one, so I was not too involved with his stand.

Bill Richardson was also having no problems with empty space. Plenty of goodies for all. Bill was also selling subscriptions to Quanta. Tony Firshman of TF Services was at the next table, and whenever I had to leave for a breath of air I had to push my way past his adoring mob. He is very strong on Hermes, the replacement processor for the 8049, reputed to get rid of that key bounce, and also to operate at higher baud rates, which is what is required for the modem operations, plus a whole lot of other improvements. I think I will get one. I believe Mechanical Affinity (USA) also have them.

Tim Swenson was in attendance with his QL Hacker's Journal, and it deals with many languages in depth. To my right was Mechanical Affinity with Paul Holmgren and Frank Davis, and every time I looked at Paul I got the impression of a bundle of $$$ in his fist. Looks more like a bookie than a trader. Lucky Paul!

John Impellizzeri and his partner were showing off their tower assembly. Looks mighty impressive to me. I could not get a price out of him for the conversion, but it appears to be less than I thought.

Then of course we had Bill Cable of Wood and Wind Computing, he that loaned me his spare QL. He was demonstrating his new program QLerk, which he has been working on for a couple of years - a program suitable to run a small business or even your household finances. I have been able to have a close look at this program, and while admitting there will be a period of getting to know QLerk, it is a very comprehensive book-keeping system.

Bob Dyl was pushing IQLR which was the sponsor of the whole show. Thanks, Bob.

## Donuts!

NESQLUG used the serving hatch in the kitchen and provided us with donuts and soft drinks, help yourself and all free! A very grateful thanks. That was also a very popular section of the show!

Who else was there? A whole bunch of dedicated QL users from a young man not yet in his teens to old fogies like myself. I have probably missed someone out, and if so, please accept my apologies and the wet noodle treatment would be appropriate. (*We hardly like to ask ...*)

After the show many of us had an excellent Dutch Treat Dinner at the Newport Beach Hotel and told tales at the tables. On Sunday, there were little meetings in the rooms of the Carlton Motel, coming and going and chatter about programs and the future and all that jazz.

Tony Firshman was running a BBS on two QLs connected by their serial ports. I could not get near, there were so many crowded into a small bedroom, but from what I could see it was a most interesting couple of hours.

I came home on Monday, but many stayed on to have a look round Cape Cod (and dodge Stuart and Tony on their bicycles.) I gather the UK people are keen to come back in 1994, and I have been told that the Germans and others would be interested in coming if these is a convention further inland.

In discussion with another trader, I asked what they thought and they said Toronto would be an ideal place. I would like to see a Sinclair convention covering ALL the Sinclair computers. This is my own thinking, and I am tossing it out for comment. The ideal place would be the Hamilton/Toronto corridor, which is easier to move around in than Toronto itself, and accommodation is cheaper. There is Queen Elizabeth Way for fast commuting right into the heart of downtown Toronto for shopping and entertainment, and a bus and train service. It's easy access from the East, and from the South it is only a few miles from highway 401.

This has been a dream with no realisation so far, but if you would come to a show in Toronto, I'm adding my address at the end of this, so write and give me your comments. And when you write, tell me what traders you would like to see there. Not all would be able to come, but we could ask them.

Was the Newport Miracle a success? I would say so. The traders were happy, and general consensus was that everyone was happy to have met so many old friends and made so many new ones. Isn't that what a convention is all about?

**Write to: Hugh H Howie, 586 Oneida Drive, Burlington, Ontario, Canada L7T 3V3.**

# QLScene

## NEW DJC CATALOGUE

Dilwyn Jones' February 1994 catalogue is now out, with several new items. DJC has agreed with **Ergon Development** of Italy to sell their software (not hardware), inlcuding Open World, Music Manager, Spectrum emulators, DEA, MasterBasic and Library Manager, etc. DJC will deal in all Ergon software apart from upgrades for existing users, which **must** be ordered direct from Italy. An information sheet with all UK and Italian currency prices is also avaiable from DJC the same as all their other leaflets - free with orders on request or with catalogues, otherwise please send an **SAE or International Reply Coupon**.

DJC's catalogue has general information as well as prices and descriptions of his wares, so is interesting reading. Dilwyn has also acquired Simon Goodwin's stock of **Speedscreen** screen accelerator, on disk (3.5in and 5.25in), microdrive and eprom. Speedscreen optimises QL display handling by replacing the QL rom code with faster routines to display text and update windows typically x5 to x10 times faster than Sinclair, Minerva or Qdos emulator code.

Speedscreen has now been tested on 68020, 030 and 040 processors as well as 68008 and 68000. Amiga emulator users can patch the code with the TAS Replacer for full compatibility. Amiga users upgrading Amiga Qdos 3.20 or earlier from a 68000 may need to repatch their Speedscreen_Code file.

A demo version is in circulation in PD and club libraries.Speedscreen costs £15 on most media, with manual, or £30 on eprom with extra documentation and disk or cartridge of add-on software. The rom is particulalry recommended for unexpanded QLs. Please state set-up when ordering. **Dilwyn Jones Computing. Tel. 0248 354023.**

## Hessler On Cue

Cueshell, a new "desktop program for all QDOS compatible systems" is out now from Albin Hessler.

Also described as a "graphically oriented **desktop** program", where options are displayed on the screen and selected with a mouse-pointer. Cueshell is based on the **Pointer Environment**, which comes as part of the package. It is intended to give easy access to everyday tasks on the QL, and at first glance at the very extensive, comprehensive, illustrated and on the whole easy-to-read instruction booklet, seems designed to do just that.

Some features are as yet little known to the QDOS environment: **Object oriented** file management (such as copying complete directory trees); "catalogue windows" which can be sized, positioned, saved and restored for any directory; up to 16 can be open at once; it is configurable from within.

Dilwyn Jones will be Cueshell's distributor in the UK and Jochen Merz in Europe. They can provide more information about it for Pointer Environment and new users - Albin's leaflet assumes quite a lot of familiarity with more advanced QL systems.

Cueshell is available on 3.5in disk for DM99 or £40. QL World will be looking for a reviewer shortly.

Albin Hessler is at **Im Zeilfeld 25, D-72631 Aichtal, Germany. Tel and fax (from UK) 010 49 7127 56280.**

# *INSTANT ACCESS - The Pocket Version*

## HARDWARE

**CL Systems** 081 459 1351
Real Time Digitizer

**Computer Technik (Jurgen Falkenburg)** 010 49 7231 81058 (Germany)
Hard disk interface, hard disk systems, tower housings for QL systems.

**Dilwyn Jones Computing (DJC)** 0248 354023
Process controller, power regulator, network prover.

**Miracle Systems** 0904 423986
Gold Card expansion card; QXL PC card; disk adapter; Extra High Density disk drives; Centronics adapter/lead; ED disks.

**Qubbesoft** PD 0376 347852
Miracle Trump Card, Expanderam, twin 3.5in disk drives. Sales and support.

**W N Richardson (EEC)** 0753 888866
Complete QL systems, monitors, keyboards and interfaces, disk drives and printers, peripherals.

**TF Services** 0344 890986
Hermes IPC, Minerva rom, keyboard membrane, repairs, spares.

## SERVICES

**Adman Services** (Dennis Briggs) 0952 255895
Spares, repairs, support, peripherals.

**Joe Atkinson** 36 Ranelagh Rd., London W5 5RJ
Roms, mdvs, spares, PD software.

**Quanta**: General Secretary: John Mason 0425 275894
User group, support, library.

**Quo Vadis** 0708 755759
QReview

## SOFTWARE

**COWO Electronic** 010 41 45 211478 (Switzerland)
QTop, Atari QL emulator, Thor support

**Deltasoft** 7 Tyrell Way, Stoke Gifford, Bristol. FlightDeck, Image D, AMD Airplan

**Digital Precision** 081 527 5493 Perfection, PC Conqueror, Lightning, Professional Publisher and many others.

**DJW Software** 0256 881701
Homebanker

**Dilwyn Jones Computing (DJC)** 0248 354023
Discover, Textidy, QL-PC Fileserver, Fleet Tactical Command, Basic Reporter, QLiberator, Filemaster, The Gopher, The Painter, Flashback, DataDesign, QPAC2 and other Pointer Environment programs.

**DI-Ren** 081 291 3751 Fleet Tactical Command (Dist. by Dilwyn Jones)

**Ergon Developments** (Davide Santachiara) 010 39 342 492323 (Italy)
ZM-X ZX Spectrum emulator, Open World, other QL sofware.

**Jochen Merz Software** 010 49 203501274 (Germany)
QL/Atari emulators, QSpread, File Finder, QPTR Pointer Environment Toolkit and other PE programs, QDesign 2, various games, and others.

**Lear Data Systems** 6 Southview Green, Bentley, Ipswich, SUffolk IP9 2DR.
PCB-CAD

**Liberation Software** 081 546 7795 QLib Basic compiler and utilities. (Dist. by Dilwyn Jones.)

**Ocean Computer Services** 061 740 9002 Professional Poolster.

**Pointer Products** 0258 455117 Pointer Environment programs

**Progs (Van Auwera)** 010 32 16 48 8952 (Belgium)
LineDesign, The Painter, The Clipart, DataDesign and others (Dist. by Dilwyn Jones; LineDesign dist. by Software87)

**Qubbesoft** PD 0376 347852
QL Home Finance, Public Domain software.

SJPD Software 0282 51854
Public Domain software

**Software 87** 33 Savernake Road, London NW3 2JU.
Text87 Plus4, Publishers' Pack.

*Some people have fax numbers on the same number as the phone. Check first.*

# Beginners' Machine Code

## In part 9, Alan Bridewell moves out of SuperBasic into EXEC jobs.

So far in this series we have been writing machine code routines which can be CALLed from SuperBasic. This is not the only way to use machine code, and is probably not the best. There are at least two alternatives.

One alternative is to write a program which can be EXECed from SuperBasic. This means that the program runs as a separate job under SuperBasic. As such, it can multi-task with other programs, or several separate copies of the same program can be running at the same time. Until compilers of SuperBasic (and other languages) came onto the market, this was the only way to write multi-tasking programs.

### What's a Job?

Before going on, we need a few words concerning jobs, for those of you who are not familiar with the concept.

When the QL is switched on, there is only one job running, and that is the SuperBasic interpreter. All the available microprocessor time is used for that job. So the QL will be ready to interpret, and execute, lines of valid SuperBasic code, which can come from the keyboard, or a RUNning SuperBasic program. Since all the available processor time is used for this, the QL cannot do anything else at the same time.

However, when a program is EXECed, it is allocated its own share of the processor time. What happens is the processor keeps swapping over between running the SuperBasic interpreter and the EXECed program. This happens over very short time intervals, to give the impression that both are working continously at the same time. Any number of jobs can be multi-tasked in this way. Of course, the more jobs that are running at the same time, the smaller the proportion of the total time each will get, so they will run slower.

Some programs do complicated things which require a lot of processor time, so any slowing down would be a disadvantage. Others, though, do very quick things, and spend most of their time waiting for input, telling them what to do next. When such programs are allocated less time it has no noticable effect. By default, all jobs running are allocated the same proportion of processor time, but this can be change. More about this another time.

A job can own input/output channels. This is why whenever we open a channel with a TRAP #2 call we have to give a job ID (usually -1 to mean "this job"). It can also own other jobs. Ownership does not mean exclusive use. Any job

LISTING 1

```
; ***************************************************************************
;                          'EXEC JOB'
; ***************************************************************************
; THIS IS HOW AN EXEC JOB SHOULD BE STARTED.
;
                    BRA.S     START       ; BRANCH TO START OF CODE
                    DC.L      0           ; (THIS IS STANDARD FORMAT FOR
                    DC.W      $4AFB       ; START OF A JOB)
                    DC.W      7           ; CHARACTER COUNT OF JOB NAME
                    DC.B      'MESSAGE'   ; NAME OF JOB
;
; OPEN THE CONSOLE CHANNEL
;
.START              LEA.L     PBLOCK,A1             ; PBLOCK ADDRESS IN A1
                    MOVEA.W   $C6,A2                ; UT_CON VECTOR IN A2
                    JSR       (A2)                  ;
;
; PRINT MESSAGE
                    LEA.L     MESSAGE,A1  ; BASE ADDRESS IN A1
                    MOVEA.W   $D0,A2      ; UT_MTEXT VECTOR IN A2
                    JSR       (A2)        ;
;
; KILL THE JOB
;
                    MOVEQ     #5,D0       ; #MT_FRJOB IN D0
                    MOVEQ     #-1,D1      ; ID OF THIS JOB IN D1
                    TRAP      #1          ;
;
; CONSOLE CHANNEL DEFINITION BLOCK.
;
.PBLOCK             DC.B      4                     ; GREEN BORDER
                    DC.B      2                     ; 2 PIXELS WIDE
                    DC.B      0                     ; BLACK PAPER/STRIP
                    DC.B      7                     ; WHITE INK
                    DC.W      150                   ; WIDTH
                    DC.W      20                    ; HEIGHT
                    DC.W      0                     ; X POSITION
                    DC.W      0                     ; Y POSITION
;
; MESSAGE.
;
.MESSAGE            DC.W      18          ; LENGTH OF MESSAGE
                    DC.B      'THIS IS A MESSAGE:'
;
; ***************************************************************************
```

can use any channel and data can be moved between jobs. All ownership means is when the particular job ends all its channels are closed automatically, and any other jobs it owns are also ended and their channels are also closed.

There are some other points that will need to be dealt with for more complicated jobs, but this will do for a start.

### EXEC Command

When a job is started with the EXEC command, the code must begin in a strict format. It starts with a JMP instruction, which should point to the code which the program runs. Although this is usually a JMP.S instruction, it could point to an address some distance away, in which case it might be a JMP.L instruction, taking 6 bytes. If a short form of the JMP instruction is used, the unused bytes must be filled with zeroes. The next two bytes contain the word $4AFB, which is a flag to tell Qdos that this is a job. This is then followed by the job name, in the form of a word giving the name length, followed by the bytes of the name itself.

Finishing a job is a little different from a CALLed routine. When a routine is CALLed from SuperBasic, we end with a RTS instruction to return to SuperBasic. This will not do for a job, because it is not running as a subroutine of another program, but as a separate job. It has nowhere to return to. What we have to do is to kill the job, so that any channels it has opened are closed, and it no longer has processor time allocated to it.

### Check Your Traps

There is a TRAP #1 call which does this called MT_FRJOB. It requires the number 5 in register D0, which tells the trap call to "force remove a job from the transient program area". (Your assembler may accept the label MT_FRJOB as the number 5.) Register D1 should contain the job ID of the job being removed: "-1" here means "remove this job". If you are ending the job because an error has occured, you may wish for the error message to be displayed, in which case the appropriate error code should go into register D3.

If we do not end with this trap call, there are various things that could happen, none of them good. The program could go on to look for more instructions beyond the last instruction, which will cause the job to crash, and quite possibly

LISTING 2

```
100 REMark Sinclair QL World HEX LOADER v 3
110 REMark by Marcus Jeffery & Simon N Goodwin
120 :
130 CLS: RESTORE :READ space:start=RESPR(space)
140 PRINT "Loading Hex...":HEX_LOAD start
150 INPUT "Save to file...";f$
160 SEXEC f$,start,byte,1024:STOP
170 :
180 DEFine FuNction DECIMAL(x)
190 RETurn CODE(h$(x))-48-7*(h$(x)>"9")
200 END DEFine DECIMAL
210 :
220 DEFine PROCedure HEX_LOAD(start)
230 byte=0:checksum=0
240 REPeat load_hex_digits
250    READ h$
260    IF h$="*":EXIT load_hex_digits
270    IF LEN(h$) MOD 2
280       PRINT "Odd number of hex digits in: ";h$
290       STOP
300    END IF
310    FOR b=1 TO LEN(h$) STEP 2
320       hb=DECIMAL(b):lb=DECIMAL(b+1)
330       IF hb<0 OR hb>15 OR lb<0 OR lb>15
340          PRINT "Illegal hex digit in: ";h$:STOP
350       END IF
360       POKE start+byte,16*hb+lb
370       checksum=checksum+16*hb+lb
380       byte=byte+1
390    END FOR b
400 END REPeat load_hex_digits
410 READ check
420 IF check<>checksum
430    PRINT "Checksum incorrect. Recheck data. ":STOP
440 END IF
450 PRINT "Checksum correct. Data entered at: ";start
460 END DEFine HEX_LOAD
470 :
480 REMark Space requirements for the machine code
490 DATA 76
500 :
510 DATA "6010": REMark              BRA.S     START
520 DATA "00000000": REMark          DC.L      #$0
530 DATA "4AFB": REMark              DC.W      #$4AFB
540 DATA "0007": REMark              DC.W      7
550 DATA "4D45553534147450": REMark   DC.B       'MESSAGE'
560 DATA "43FA0018": REMark          LEA.L     PBLOCK,A1
570 DATA "347800C6": REMark          MOVEA.W   $C6,A2
580 DATA "4E92": REMark              JSR       (A2)
590 DATA "43FA001A": REMark          LEA.L     MESSAGE,A1
600 DATA "347800D0": REMark          MOVEA.W   $D0,A2
610 DATA "4E92": REMark              JSR       (A2)
620 DATA "7005": REMark              MOVEQ     #5,D0
630 DATA "72FF": REMark              MOVEQ     #-1,D1
640 DATA "4E41": REMark              TRAP      #1
650 DATA "04": REMark                DC.B      4
660 DATA "02": REMark                DC.B      2
670 DATA "00": REMark                DC.B      0
680 DATA "07": REMark                DC.B      7
690 DATA "0096": REMark              DC.W      $96
700 DATA "0014": REMark              DC.W      $14
710 DATA "0000": REMark              DC.W      0
720 DATA "0000": REMark              DC.W      0
730 DATA "0012": REMark              DC.W      $12
740 DATA "5448495320495320412041204D4553534147453A"
750 REMark                                     'THIS IS A MESSAGE'
760 DATA "*",4877
```

will cause the whole QL to crash. The program could go into a closed loop of instructions, forever doing nothing over and over again, and using processor time for nothing. It could reach a point where it was waiting for input which never comes, again using processor time doing nothing. Clearly, if we have finished with the job, it is best to kill it properly.

What we have here is the essential minimum for a successful job, a proper start, and a proper end. What comes in between can be whatever we want the job to do.

To illustrate the point **Listing One** is a minimal job. (Clearly we could have a job which did nothing at all, but then it would be a bit difficult to show it had worked. This job will at least produce something on the screen to show us it has been done.)

We begin with the standard format for the start of a job. Note the line:

```
DC.L 0
```

which puts in four zeroes to fill up the space because BRA.S only uses two of the six bytes (which would have been needed if we started with a BRA.L instruction.

Next comes the program proper. It opens a console channel, and then prints some text in it. This has all been covered in previous parts of this series.

Only two points need making. First, there is no need to put the channel ID into register D1 for the UT_MTEXT subroutine, because it will already be there, left by the UT_CON trap call immediately before. Second, there is no need to deliberately close the console channel because it will be close by the MT_FRJOB trap call at the end.

The listing ends with the parameter block for the console and the message to be displayed.

### SEXEC!

When you assemble the listing, it is important to remember that the code has to be saved using the SEXEC command and not the SBYTES ccmmand, because SuperBasic has to recognize it as a file suitable for the EXEC command.

**Listing Two** contains the code in Marcus and Simon's Hex Loader for those who do not have an assembler, with the slight difference that line 160 has been altered to give the SEXEC command rather than SBYTES.

Once you have got the program to EXEC properly, you can try experimenting with it using any of the ideas covered so far in this series. None of them should give any particular problems as long as you avoid using the stack. (Each job must have its own stack, but that another story.) The only other thing to realise is this. If your job is going to get input from the keyboard, you will have to used Ctrl-C to get between the job's flashing cursor to SuperBasic's channel #0 flashing cursor.

The program in Listing One is so short and is completed so quickly that it hardly rates as a program for multi-tasking. But by writing longer routines, with multiple loops so that they take an appreciable time to complete, it is possible to EXEC a second copy while the first is still running and see the effect. You could also EXEC Listing One while a longer job was running, so that you could see it completed without having any noticable influence on the other job.

Essentially, there is nothing difficult about EXEC routines once a few ground rules have been understood. Simple EXEC routines are no more difficult to write than simple CALL routines.

Next time we shall look at some other aspects of EXEC job programming.

# CLUB ACCESS

**AUSTRIA**
**DER Computer Club.** Contact: Peter Postl, Stiegerg 5, 1150 Wien, Austria.
**BELGIUM**
**Club Sinclair BruQsl.** Contact: Jaques Tasset, Aarlenstraat 104, 1040 Brussels, Belgium.
**QL Club.** Contact: Leon Thianche, Rue Paul Wemaere 12-14, 1150 Bruxelles, Belgium.
**FRANCE**
**QL Contact France.** Contact: Jean-Louis Dianoux, 22 Avenue Lenine, 93230 Romainville, France.
**GERMANY**
**Sinclair QL User Club eV.** Contact: Franz Herrmann, Talstrasse 21, d-W5460 Ochenfels, West Germany. Magazine: Quasar.
**GREECE**
**QL Athens Club.** Contact: Stathis Grigoriadis, Tarsu 6-8, 10440 Athens, Greece.
**HOLLAND**
**Sin_QL_Air.** Contact: Secretary: Marco Holmer, J P Coenstraat, 3531 EN Utrecht, Netherlands. Magazine: Quasar. Editor: Same as Secretary. (Chairman: Cor Biemans, Elzenstraat 5, 5461 CL Vehgel, Netherlands.
**ITALY**
**Qitaly Club.** Contact: Roberto Orlandi, Via Brescia 26, 25039 Traveglianto (BS), Italy. Tel. +39 30 6863311. Magazine: Qitaly Magazine. Editor: Dr Eros Forenzi, Via Valeriana 44, 23010 Berbenno (SO), Italy. Tel. +39 342 590450.
**NORWAY**
**Norwegian AllSinclair Association (NASA).** Contact: P Monstad, NASA, N-5580 Oelen, Norway. Magazine: Sinclair Magazine.
**SCOTLAND**
**Scottish QL Users Group.** Contact: Alan Pemberton, 65 Lingerwood Road, Newtongrange, Midlothian EH22 4QQ. Newsletter
**SPAIN**
**Qliper.** Editor: Salvador Merino, Ctra C_diz, Cer_micas Mary, 29640 Torreblanca del Sol, Spain. **Magazine: Qliper.**
**SWEDEN**
**Svenska QL-Gruppen.** Contact: Johan Boman, Toftaasgatan 73, 42147 Vastra Frolunda, Sweden. Magazine: QL-Bladet.
**Jamten-TCL** (International QL echomail conferences) BBS +46-63-133330. Michael Cronsten, Radhusgatan 61E, 83137 Ostersund, Sweden.
**TURKEY**
**QL Qlub.** Contact: Bulent Artuz, Prof. Sitesi B/1 D/5, Etiler 80600, Istanbul, Turkey.
**UK**
**Quanta (UK):** Membership Secretary, Bill Newell, 213 Manor Road, Benfleet, Essex SS7 4JD. Magazine: Quanta. Editor: Bill Fuggle, 20 Widnes Avenue, Selly Oak, Birmingham B29 6QE. Quanta.
**Ashstead sub-group:** Derek Stewart, 13 Beales Road, Great Bookham, Surrey .
**Bristol sub-group:** Roy Brereton, 94 Teignmouth Road, Clevedon, Avon.
**East Anglia:** Geraint Jones, 1 Bridgate Court, Thetford IP24 3AJ. (Also market Rom Switcher.)
**Essex sub-group:** Dave Walker, 22 Kempton's Mead, Potters Bar, Herts EM6 3HZ.
**London sub-group:** Jeremy Davis, 6 Elmcroft Crescent, Harrow, Middx. HA2 6HN.
**North-east:** Derek Stewart, 20 Emily Street, Gateshead.
**Northern Ireland:** Billy Turkington, Fairyhill, Rostrevor, Newry, Co. Down.
**Mid-Southern** sub-group: Geoff Fish, 44 Billing Avenue, Wokingham, Berks.
**QL User Group (West Midlands):** Mike Bedford-White, 16 Westfield Road, Acocks Green, Birmingham.
**QL MUG (Merseyside):** G Reynolds 051 932 1484, evenings. Meet fortnightly.
**Solent sub-group:** Graham Evans, 32 Reeves Way, Lowford, Bursledon, Southampton. Tel. 0703 403350.

SINCLAIR QL WORLD

# SUBSCRIBE TO SINCLAIR QL WORLD AT LAST YEARS PRICES AND SAVE £££££'s

£££££

**UK**
**£23.40**
**(12 MONTHS)**

**EUROPE**
**£32.90**
**(12MONTHS)**

**REST OF WORLD**
**£40.90**
**(12 MONTHS)**

PLEASE SEND YOUR SUBSCRIPTION FORM TO SINCLAIR QL WORLD, THE BLUE BARN, TEW LANE, WOOTTON, WOODSTOCK, OXON. OX71HA.
TEL:0993 811181
FAX:0993 811481

I WISH TO SUBSCRIBE TO SINCLAIR QL WORLD
WE HAVE HAD TO RAISE THE COVER PRICE OF QL WORLD BUT WILL BE HOLDING LAST YEARS SUBSCRIPTION PRICE FOR THE IMMEDIATE FUTURE ONLY.

**Existing subscribers will not be affected.**

PLEASE DEBIT MY CREDIT CARD ACCESS/VISA

CARD NUMBER.....

EXPIRY DATE

PLEASE SEND ME A 12 MONTH SUBSCRIPTION TO SINCLAIR QL WORLD.

NAME.....

ADDRESS.....

.....

TELEPHONE NUMBER.....

SIGNED.....

DATE.....

STARTING WITH ISSUE.....

# MICRO ADS

**SINCLAIR QL** + twin floppy drives, 256k memory, separate keyboard, many games £125 0491 681821

**QL JM EXPANDED,** SUPER-Q board 512k, 102-key keyboard, NEC 3.5" drive, Philips monitor, Epson RX-80 printer, Psion V2.35, QRAM, QUIMI, Manual, DL World Mags, £200 ono, Tel:0952 676668 evening.

**QL plus** twin three and a half inch drives. 768K Trump Card. Philips RGB input monitor. Borther M1009 printer. Software inclusing Turbo compiler. Manuals. #120 or offers. Sussex 0243 531402. Also 512K Expanderam.

**JM/JS QLs**, manuals, extra RAMs, twin floppy, two monitors, Canon 1080A, cables, Psion software, Turbo Quill Plus, FLight Simulator, Archivist, Lightning, Archive tutor, Chess etc. Books, magazines. #250 SAE for list. 021 430 7772.

**QL unexpanded JM** version, shinwa CPA 80Q printer, joystick, Software including Psion suite, Eye-Q and Games, 40 MDV's, Manuals etc, £150 Tel:0256 764006 evenings only

**QL Minerva Rom,** Trump Card 3-1/2" D/D,Miracle centronics (x2) Tandata modem,serial printer, 6 years QLWorld, Quanta mags, books, software, b/w television extra MDV. Sensible offers please. Tel: 0977 515428

**COMPLETE OUTFIT** QL 640k js. Twuin Cumana 5.25" disc drives. 14" colour monitor. Tandata modems. Manneshan MT8. Printer lightening S.E. Backup QL manuals etc. All VGWO offers Cliff 0934 812640.

***Professional poolster REVOLUTIONARY NEW SYSTEM***

***COMBINES***

*Sound logic with A.I. to win £ hundreds £*
*Analysis of Bookies Factors for matches*
*Common sense improves chances of Mega-Win*
*UNIQUE ENTRY SYSTEM*
*Takes only a few minutes each week*
*Over 300 teams in data-bank.*
*Minor league teams automatically added.*
*Prior to PC release @ £99.00*
*To QL users offered at SPECIAL PRICE £49*

***THIS MONTH***

*MONEY BACK GUARANTEE - Make a profit in your first full season or your money back.*
*256K ROM minimum. Discs Only. State 3.5 or 5.25*

***CHEQUE OR P.O. TO S.CANTON, OCEAN COMPUTER SERVICES, PO BOX 68, NORTH DO. Manchester M8 6NJ***
***Tel. 061 740 9002***

# CLASSIFIED

Please send coupon to: Advertising Department Sinclair/QL World, The Blue Barn, Wootton, Woodstock, Oxon. OX7 1HA. Please place this Classified Advertisement in the next issue of Sincair/QL world.

NAME .....

ADDRESS .....

.....

DAYTIME TELEPHONE NO .....

I enclose a cheque/P.O. to the value of £17.65 (inc. VAT) for a lineage advertisement of no more than 30 words. Please make cheques payable to: Arcwind Ltd.

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

# DJC Dilwyn Jones Computing

41 BRO EMRYS, TAL-Y-BONT, BANGOR,
GWYNEDD, LL57 3YT, GREAT BRITAIN
FAX/TEL: (0248) 354023

## DATABASE SOFTWARE

COCKTAILS WAITER£10 00
DATA DESIGN 3 ..... £60.00
DATA DESIGN API ..... £20.00
DBEASY ..... £15.00
DBPROGS ..... £15.00
FLASHBACK ..... £25.00
FLASHBACK SE ..... £40.00
SUPER DISK INDEXER ..... £12.00

## DTP & CLIPART

PAGE DESIGNER 3 ..... £40.00
Available at last!
UPGRADE FROM PD2 ..... £25.00
SCANNED CLIPART 1 (U) ..... £10.00
SCANNED CLIPART 2 (U) ..... £10.00
THE CLIPART (U) ..... £12.00

## FILE HANDLING

FILEMASTER ..... £12.00
FILES 2 (U) ..... £12.00
4MATTER & LOCKSMITHE ..... £23.50
LOCKSMITHE (U) ..... £14.95
MDV TOOLCHEST ..... £14.95
THE GOPHER (U) ..... £12.00
WINBACK 2 ..... £25.00

## FILE TRANSFER

CONVERT-PCX ..... £10.00
DISCOVER ..... £20.00
MULTI-DISCOVER ..... £30.00
OPD INTERCHANGE ..... £15.00
QL-PC FILESERVER ..... £24.50
SToQL ..... £24.50
TEXTIDY ..... £15.00

## GAMES & LEISURE

FIVE GAME PACK (U) ..... £12.50
FLEET TACTICAL COMMAND (U) ..... £39.95
FTC DATA PRINTER ..... £9.95
FLIGHTDECK (U) ..... £15.00
FUGITIVE ..... £9.95
GREYWOLF ..... £12.50
OPEN GOLF ..... £12.50
QUESTION MASTER (U) ..... £10.00
QUIZ MASTER 2 (U) ..... £12.50
RETURN TO EDEN ..... £17.50
SOLITUDE (U) ..... £15.00
SQUIDGY ROUND THE WORLD (U) ..... £12.50

## LABELLING

ADDRESS BOOK&LABEL PRINTER ..... £15.00
SUPER DISK LABELLER ..... £10.00

## GENEALOGY

GENEALOGIST 3 ..... £60.00
new pointer driven version
Upgrade 2nd Edition ..... £33.00
upgrade other versions, ask
GENEALOGIST 2ND EDITION ..... £30.00
BUDGET 128K GENEALOGIST (U ..... £12.00

## GRAPHICS

IMAGE-D (U) ..... £10.00
IMAGE PROCESSOR ..... £15.00
LINE DESIGN ..... £100.00
PAINTER ..... £25.00
PICTUREMASTER ..... £15.00
PICTUREMASTER PLUS ..... £20.00
QRACTAL ..... £20.00
QUICK MANDELBROT 3 (U) ..... £15.00
SCREEN COMPRESSION ..... £10.00
SCREEN SNATCHER (U) ..... £10.00
SIMPLE VIDEO TITLES ..... £5.00
TRANS24 (U) ..... £10.00

## HARDWARE

MICRO PROCESS CONTROLLER
new redesigned version ..... £64.50
MPC SOFTWARE ..... £9.95
NETWORK PROVER ..... £4.00
QPOWER REGULATOR ..... £24.95
(unsuitable Gold Card)
SERMOUSE ..... £40.00
VOICE ANALYSER-New!
Ask for details and price
Add £2 50 postage for MPC,
SERMOUSE AND VOICE ANALYSER

## MAGAZINES

QREVIEW ..... £2.00
QL TECHNICAL REVIEW ..... £2.00
issues 1-8 available
QL ADVENTURER'S FORUM ..... £1.75
Issues 1-9 available
QL LEISURE REVIEW ..... £2.00
issues 1-2 available
(UK prices only shown)

## PROGRAMMING

BASIC REPORTER (U) ..... £10.00
DISA ..... £29.00
DJTOOLKIT (U) ..... £10.00
EASYPTR 3 PART 1 ..... £40.50
EASYPTR 3 PART 2 ..... £20.00
EASYPTR 3 PART 3 ..... £20.00
MEGATOOLKIT DISK (U) ..... £25.00
MEGATOOLKIT EPROM (U) ..... £40.00
QLIBERATOR 3 36 ..... £50.00
BUDGET QLIBERATOR (U) ..... £25.00
QLOAD AND QREF (U) ..... £15.00
S EDIT EDITOR ..... £20.00

## SCREEN DUMPS

SIDEWINDER PLUS ..... £24.95

## SPREADSHEET PRINTING

SIDEWRITER (U) ..... £15.00
3D TERRAIN ..... £12.50

## SUNDRIES

3 5" DSDD DISKS each ..... £0.40
3 5" DSHD DISKS each ..... £0.70
3 5" DISK LABELS (roll 100) ..... £2.00
3 5" LABELS (printer roll) ..... £2.50
ADDRESS LABELS (roll 100) ..... £2.00
MICRODRIVE CARTRIDGES ..... £2.50
MDV LABELS (roll 100) ..... £2.00
MOUSE MAT ..... £2.50
3 5" DISK DIVIDERS (20) ..... £3.00
Add £0 50 postage for labels or mouse
mat if only ordering those, or £2 50 postage
for disks or disk box divider sets

## TEXT

BANTER BANNERS ..... £25.00
BIBLE TEXT ..... £20.00
QTYP2 ..... £29.95
QUICK POSTERS (U) ..... £10.00
ROB ROY PACK (U) ..... £10.00
SPELLBOUND ..... £30.00
SPELLBOUND S E ..... £50.00
TEXTIDY ..... £15.00
TEXT 'N' GRAPHIX ..... £20.00

## OTHER SOFTWARE

HOME BUDGET (U) ..... £20.00
PRINTERMASTER (U) ..... £20.00
QPAC1 ..... £19.95
QPAC2 ..... £39.95
THE Pointer Environment package!
QTOP ..... £29.95
SCREEN DAZZLER ..... £15.00
SCREEN ECONOMISER (U) ..... £10.00
SLOWGOLD (U) ..... £5.00
SPEEDSCREEN (U) ..... £15.00
SPEEDSCREEN EPROM (U) ..... £30.00
TASKMASTER ..... £25.00
VISION MIXER 1 ..... £10.00
VISION MIXER PLUS ..... £22.50

CALL OR WRITE FOR A FREE COPY OF
OUR QL SOFTWARE CATALOGUE
DETAILING AROUND 100 QL PRODUCTS

PLEASE NOTE: (U) ABOVE MEANS THAT
THE SOFTWARE IS SUITABLE FOR USE
ON A 128K QL

TERMS: POSTAGE-software sent post free to UK, overseas add £1 00 per program (maximum £3 00) Floppy disks, SERmouse, etc add £2 50 postage (see above) PAYMENT - IN UK currency (pounds sterling) only, please Valid methods of payment are cheque drawn on UK branch of bank or building society, Eurocheque, Postal Order, cash (send by registered post) or by credit card - Visa, Access, Mastercard and Eurocard accepted Please make cheques etc payable to DILWYN JONES COMPUTING Minimum order value £5 00 (due to bank charges) Goods remain property of Dilwyn Jones Computing until paid for in full Orders normally sent out within a few days of receipt, we will attempt to advise if any delay anticipated due to stock problems Orders can be accepted by telephone if paid for with credit card Orders paid with credit card can only be sent to cardholder's address under card company rules
FAX - We now have a Fax machine on our usual number (0248) 354023

VISA
Master Card
Euro Card